MACHINE CONSCIOUS ARCHITECTURE FOR STATE EXPLOITATION AND DECISION MAKING

DISSERTATION

Mark M. Derriso, USAF, Civilian

AFIT-ENG-DS-13-M-01

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government. This material is declared a work of the United States Government and is not subject to copyright protection in the United States.

AFIT-ENG-DS-13-M-01

MACHINE CONSCIOUS ARCHITECTURE FOR STATE EXPLOITATION AND DECISION MAKING

DISSERTATION

Presented to the Faculty

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

Mark M. Derriso, BS, MS

Civilian, USAF

March 2013

AFIT-ENG-DS-13-M-01

MACHINE CONSCIOUS ARCHITECTURE FOR STATE EXPLOITATION AND DECISION MAKING

Mark M. Derriso, BS, MS
Civilian, USAF

Approved:

Richard A. Raines — 1 Mar 13
Richard A. Raines, Ph.D. (Chairman) — Date

Mark E. Oxley — 5 Mar 13
Mark E. Oxley, Ph.D. (Member) — Date

5 Mar 13
Steven K. Rogers, Ph.D. (Member) — Date

Signed:

Heidi R. Ries — 11 Mar 2013
HEIDI R. RIES, Ph.D.
Interim Dean, Graduate School of Engineering
and Management — Date

AFIT-ENG-DS-13-M-01

Abstract

This research addressed a critical limitation in the area of computational intelligence by developing a general purpose architecture for information processing and decision making. Traditional computational intelligence methods are best suited for well-defined problems with extensive, long-term knowledge of the environmental and operational conditions the system will encounter during operation. These traditional approaches typically generate quick answers (i.e., reflexive responses) using pattern recognition methods. Most pattern recognition techniques are static processes that consist of a predefined series of computations. For these pattern recognition approaches to be effective, training data are required from all anticipated environments and operating conditions.

The proposed framework, Conscious Architecture for State Exploitation (CASE), is a general purpose architecture designed to mimic key characteristics of human information processing. CASE combines low- and high-level cognitive processes into a common framework to enable goal-based decision making. The CASE approach is to generate artificial phenomenal states (i.e., generate qualia = consciousness) into a shared computational process to enhance goal-based decision making and adaptation. That is, this approach allows for the appropriate decision and corresponding adaptive behavior as the goals and environmental factors change.

To demonstrate the engineering advantages of CASE, it was used in an airframe application to autonomously monitor the integrity of a flight critical structural component. In this demonstration, CASE automatically generated a timely maintenance recommendation when unacceptable cracking was detected. Over the lifetime of the investigated component, operational availability increased by a minimum of 10.7%, operational cost decreased by 79%, and maintenance intervals (i.e., *MTBM*) increased by a minimum of 900%.

I would like to dedicate the work performed in this dissertation to my parents who taught me from childhood never to give up on my dreams. I also would like to thank the rest of my family and friends for their patience and support during this journey. Especially, I would like to thank my wife, who has been extremely supportive and patient during all my educational endeavors over the years. I am very grateful to have you in my life.

Acknowledgments

I thank my committee and advisor for their guidance and wisdom. Without them, this research would not have been possible. In addition, I thank all my colleagues who supported me during this endeavor. I am extremely appreciative.

Mark M. Derriso

Table of Contents

List of Figures

Figure Page

List of Tables

Table Page

MACHINE CONSCIOUS ARCHITECTURE FOR STATE EXPLOITATION AND DECISION MAKING

I. Introduction

1.1 Background

During the 1980's, a major breakthrough in computational intelligence occurred with the development of training rules for hidden units in artificial neural networks (ANNs). ANNs mimic a significant aspect of the architecture of biological neural networks – that is, brains – as a large number of densely interconnected, but fundamentally simple processing elements, neurons. Although artificial neurons are only coarse approximations of biological neurons, and ANNs are still coarser approximations to brains, ANNs have been used successfully across a wide range of applications, from medical image diagnosis to loan application processing [18] [27]. Problem areas, such as automatic target recognition (ATR), malware detection and structural health monitoring (SHM) have not been successfully solved with current ANN approaches [6] [13] [16] [28]. The remaining problems typically involve high degrees of uncertainty and a dynamic environment. A new computational approach is needed to address these classes of problems.

The next major advancement in computational intelligence will result from extending the mimicry from the neuronal level to the cognitive architecture level of the brain. Conscious architecture refers to how information flows through and is controlled in the brain [31]. According to Morsella, an essential feature of human brain processing is that multiple simultaneous, low-level unconscious processing modules provide inputs to a higher-level conscious processing module, where the conscious processing must disambiguate conflicting objectives [22]. For example, imagine reaching with your bare hand into the microwave to pull out a just-heated dinner plate. If the pain of the hot plate does not register until you have carried it halfway across the kitchen, you may not realize it at the time, but your brain is disambiguating two conflicting objectives. One unconscious reflexive process is demanding the minimization of tissue damage, and therefore suggests dropping the plate. A conflicting unconscious process is demanding the satisfaction of hunger, by endurance of pain, which will shortly lead to satiation of hunger. From an engineering standpoint, just as it was not necessary to know and closely replicate the inner workings of a neuron to achieve practical benefits from an ANN, we believe that the next level of computational intelligence will require duplicating the processes which are combined to control the musculoskeletal system in light of conflicting objectives. However, we believe significant engineering advantage can be achieved by providing a computational pathway for considering, and then resolving the unconscious and conscious aspects of decision making.

Our proposed architecture, Conscious Architecture for State Exploitation (CASE), combines low- and high-level cognitive processes into a common framework to enable goal-based decision making. The CASE approach is to generate artificial phenomenal states (i.e., generate qualia = consciousness) into a shared computational process to enhance goal-based decision making and adaptation. That is, this approach allows for the appropriate decision and corresponding adaptive behavior as the goals and environmental factors change. Given the current state and the desired goal, CASE recommends an appropriate action to take for achieving the given goal. This architecture, which exploits phenomenal states, is shown to improve the performance of computational intelligence for applications currently unsolved using traditional approaches.

1.2 Research Goals

The goals of this research are to design and develop the CASE framework, and then demonstrate it in an application for which traditional approaches have been inadequate. The success of the proposed approach will be based on the engineering advantage using the philosophy incorporated in CASE.

1.3 Document Overview

The remainder of this document is organized as follows. Chapter 2 provides a background on designing conscious systems and their associated characteristics. A conscious framework and investigated conscious architectures are discussed as well. Chapter 3 describes the overall functions and processing philosophy of CASE, along with an illustration of how CASE incorporates key characteristics of consciousness. CASE is demonstrated in an SHM application in Chapter 4, and the results are compared with the current practice for ensuring structural integrity. A summary of this work follows in Chapter 5, highlighting the contributions and impact of this research. Additionally, areas for research extension are also explored.

II. Literature Review

2.1 Overview

This chapter provides the background research and knowledge of this research. It covers theories of consciousness, machine conscious architectures and characteristics of a conscious framework. In addition, functional and phenomenal tenets of consciousness are discussed.

2.2 Consciousness Theory

One of the oldest and most widely-cited theories of cognition that includes conscious and subconscious aspects of its representation is the Global Workspace Theory (GWT). GWT was developed by Bernard Baars to qualitatively account for a large set of conscious and unconscious processes [3]. The principle theory behind this model is the information flow of multiple parallel, specialized processes that compete and cooperate for access to the global workspace (i.e., working memory).

The most relevant information from these specialized processes is allowed access to the global workspace to compete for processing attention. The global workspace represents the process through which information is integrated and processed before a conscious state is determined. Within the global workspace is a spotlight that represents the focus of attention. The global workspace offers the flexibility to view content under the spotlight at different levels of abstraction similar to a zoom lens. Only those aspects of the working memory that are within the 'spotlight' are conscious (i.e., qualia). Context information (e.g., goals) is provided to the global workspace to influence the content under the spotlight. This conscious information is then broadcast back out to the specialized processes to possibly change their state if appropriate.

Figure 1 depicts the global workspace architecture. The five external circles in Figure 1 symbolize the unconscious parallel processes, and the rectangle represents the global workspace (i.e., working memory). These parallel processes compete for entry into the global workspace. The content of a specific parallel process enters the global workspace only when it wins the competition. The circle inside the rectangle (i.e., working memory) is the "spotlight" which represents the focus of attention. Content within the global workspace only enters the "spotlight" when it is attended to. Since Baars only gives a metaphorical description of the GWT, more engineering design is required before this architecture can be realized.

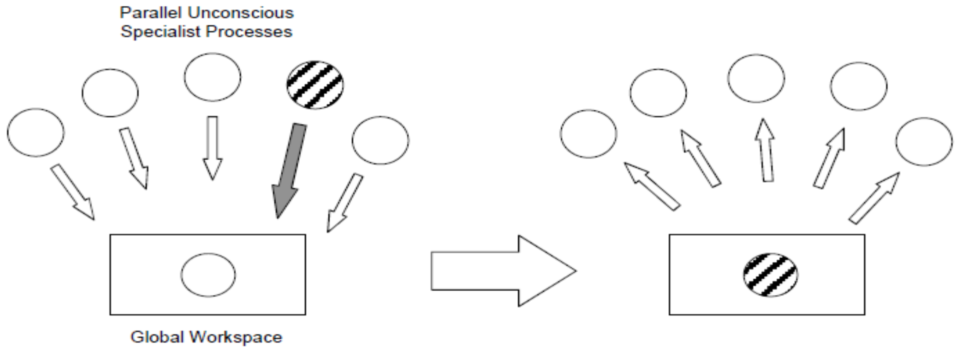


Figure 1. The Global Workspace Architecture [29]

Several Machine Conscious (MC) researchers have engineered cognitive models inspired by the GWT for their specific investigations [2] [4] [29] [30]. Arrabales developed a cognitive architecture inspired by the GWT to explore a plausible functional explanation of how conscious experience could be generated from a global workspace [2]. Specifically, Arrabales' research focused on characterizing artificial qualia (i.e., subjective experience) as the contents that appear under the global workspace "spotlight." This research primarily focused on the functionality of MC with no claims made toward the generation of phenomenal states. The GWT-based architecture was developed to investigate visual qualia in a robotic application. This MC model was designed to identify explicit contents (i.e., overt perception) of an autonomous robot. The MC model was implemented using the cognitive architecture CERA and the functional model CRANIUM [2]. A diagram of this computational model is depicted in Figure 2.

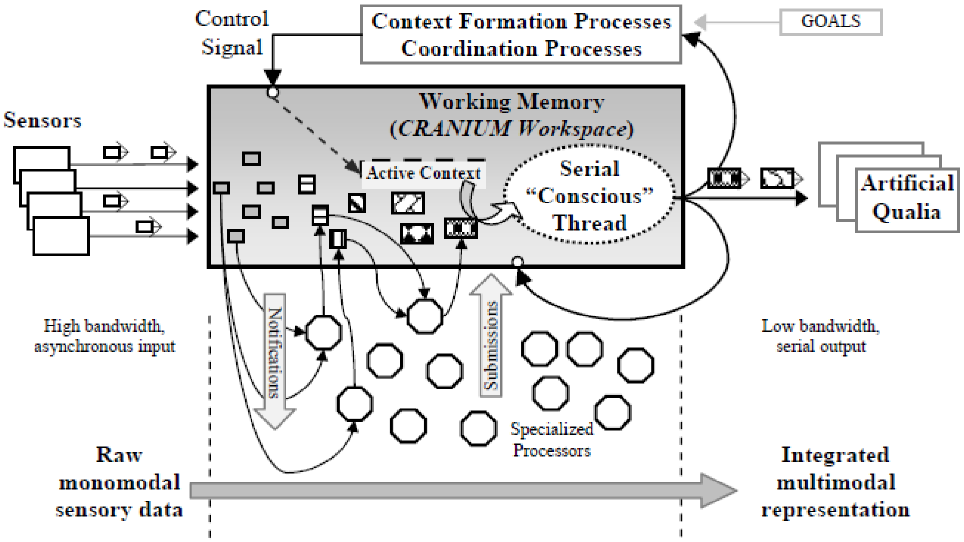


Figure 2. CRANIUM Computational Model [2]

Murray Shanahan also developed a model inspired by the GWT for his research on MC [29]. The objectives of Shanahan's architecture were to demonstrate the simulation hypothesis (i.e., thoughts are "internally simulated interaction with the environment") and prove that it is feasible to use the theory of consciousness for controlling robots. Shanahan used weightless neurons, G-RAM (Generalizing Random Access Memories), for implementing the unconscious processing functions of the architecture. These neurons employed single-shot training for which the update function can be rapidly computed. The conscious processing element of the architecture was realized via an internal simulation for cognitive functions such as anticipation and planning. This architecture was implemented to control a simulated robot.

The GWT provides a good foundation for developing functionally conscious systems as demonstrated by Arrabales, and Shanahan. However, the “hard problem” of creating artificial phenomenal states, which is an essential element of consciousness, has not been adequately addressed during these investigations. A general purpose GWT-inspired framework (i.e., architecture) that incorporates both functional and phenomenal conscious elements has yet to be achieved.

Since the current research is focused on developing a general purpose MC architecture for generating phenomenal states, we begin the next section by discussing the recommended attributes of a conscious/phenomenal state framework.

2.3 Conscious Framework

Developing a framework for consciousness is required in order to investigate the “hard problem” of phenomenal states. Several researchers such as Crick and Koch, and Rogers and Kabrisky have described a framework for consciousness [8] [28]. Crick and Koch consider a framework – not a detailed hypothesis or set of hypotheses, but a proposed method for attacking a scientific problem, often suggesting testable hypotheses. Crick and Koch also believe that a good framework is one that appears reasonably plausible given the available scientific data and produces reasonably accurate results. A list of features recommended by Crick and Koch for developing a conscious framework is given below [8].

1. **Unconsciousness –** Humans are not directly conscious of their thoughts, but a sensory representation of them in their imagination.
2. **Consciousness –** Many actions in response to sensory inputs are rapid, transient, stereotyped and unconscious. Consciousness deals more slowly with broader, less stereotyped aspects of the sensory inputs and takes time to decide on appropriate thoughts and responses.
3. **Coalitions of neurons –** Neurons that form coalitions to support one another and compete among the other coalitions.
4. **Explicit representations –** A small set of neurons exists that responds as a detector for that feature, without further complex neural processing.
5. **The higher level first –** For a new visual input, the neural activity first travels rapidly and unconsciously up the visual hierarchy to a high level (this might instantiate zombie mode).
6. **Driving and modulating connections –** Connections to cortical neuron fall into two broad classes: driving and modulating inputs.
7. **Snapshots –** Conscious awareness (for vision) is a series of static snapshots with ‘motion’ painted on them. That is, perception occurs in discrete epochs.
8. **Attention and binding –** Attention can usefully be divided into two forms: either rapid, saliency-driven and bottom-up or slower, volitionally controlled and top-down.

9. **Styles of firing –** Synchronized firing may increase the effectiveness of a neuron, while not necessarily altering its average firing rate.
10. **Penumbra and meaning** – Visual features interest such a set of neurons, but how does the brain know what that firing represents? This is the problem of meaning.

This list of conscious characteristics is not all-inclusive, as it is not feasible to duplicate the human information process entirely. However, we do believe that it is possible to mimic selected features of human consciousness for providing some engineering advantages in problems currently unsolvable with classical methods.

Most MC investigations develop conscious models inspired by the GWT for a specified purpose as described earlier; however, none of them explicitly demonstrated any of the suggested attributes of consciousness given above. This research identifies a selected set of the conscious attributes listed above and incorporates them in the proposed architecture. This research demonstrates that these attributes, used as guiding principles, are critical for realizing a general purpose MC architecture. From these fundamental attributes, the functions of consciousness are discussed.

2.4 Conscious Functions

Phenomenal states are often referred to as “subjective experience,” “qualia,” “sentience,” “consciousness,” and “awareness” [22]. This document uses the terms consciousness and phenomenal states synonymously. The functional role of phenomenal states (i.e., consciousness – the generation of those phenomenal states, qualia) still remains one of the greatest challenges for psychological science [22]. According to

Aleksander, a phenomenal system is one that is responsible for the behavior of the system by reflecting the properties of the real world [1]. In addition, Baars believes that phenomenal states allow for the global access of information (e.g., auditory, affective and visual information) [3]. In this document, we take the position proposed by Clark that phenomenal states are useful for the reason-and-memory-based selection of action, which uses knowledge from different bases that requires integration [7]. This concept is referred to as the *integration consensus* [22].

According to Morsella, an essential feature of human cognition is that multiple simultaneous, low-level processing modules provide inputs to a higher level of processing, where the higher-level processing must disambiguate conflicting objectives. This process can be referred to as the Supramodular Interaction Theory (SIT) [22].

SIT proposes that phenomenal states play a critical role in permitting interactions among a variety of response systems (i.e., modules) with different objectives. Without phenomenal states, output from these different systems would be incapable of collectively influencing action [22]. Figure 3 depicts the SIT. Response System A is concerned with how the organism should physically interact with the world. Response System B is an incentive system concerned with whether the organism should approach or avoid a stimulus. The output of these response systems only interacts in the phenomenal field, and they modulate a different aspect of the phenomenal experience.

These phenomenal experiences (i.e., subjective experiences) are internally displayed using a phenomenal representation (i.e., mental states). Phenomenal representations are not a completely accurate portrayal of the world, but should be capable of generating a stable, consistent and useful depiction of the environment suitable for successful decision making.

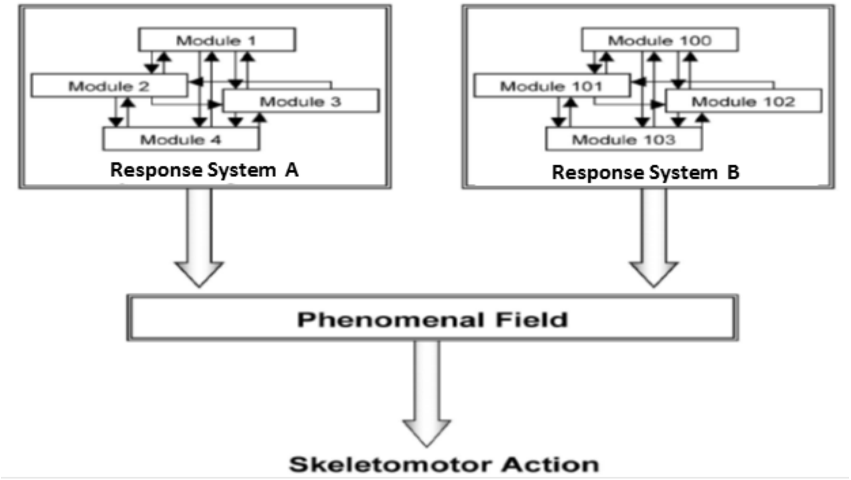


Figure 3. Supramodular Interaction Theory [22]

Morsella provides the tenets of SIT, which are listed below:

1. Phenomenal States allow information from diverse sources to interact in order to produce adaptive action.
2. Relatively few kinds of information require conscious interaction because many kinds of information can interact unconsciously.
3. Phenomenal states are required for outputs of different supramodular response systems to interact.
4. Interactive processes occurring among modules within response systems can be unconscious, but interactive processes across systems require conscious processing.

5. The response tendencies of response systems may conflict with skeletal muscle plans.
6. The outputs of response systems incessantly modulate the phenomenal field, regardless of whether there is conflict.
7. Without phenomenal states, the outputs of the different systems would be encapsulated and incapable of collectively influencing action.

As discussed in the previous section regarding the importance of the conscious attributes, the conscious functional tenets are just as essential when developing a general purpose framework. In fact, this research shows that the two are coupled, and to exhibit any of the given conscious attributes requires one or more of the tenets for functionality. Consequently, we will also demonstrate a subset of the functional tenets listed above via our proposed architecture.

The next section briefly discusses the characteristics of phenomenal states (i.e., qualia) and their associated representation in MC applications as proposed by Rogers et al. [28]. In addition, Rogers et al. provide a list of recommended tenets for designing a general purpose MC architecture.

2.5 Qualia Exploitation of Sensor Technology

Rogers et al. discussed the potential benefits of using MC systems in military applications [28]. Specifically, they discussed how MC methodologies could potentially improve the capability of Automatic Target Recognition (ATR) techniques that are currently unable to meet the warfighter’s needs. To improve the ATR capabilities (e.g., combat identification), Rogers et al. suggest developing a general purpose machine-based recognizer called QUalia Exploitation of Sensor Technology (QUEST) [28]. The objective of QUEST is to construct a subjective representation (i.e., phenomenal states) to improve the characterization of entities in the environment. The QUEST approach for developing a qualia-based system is to establish a list of guiding tenets to serve as the fundamental driving characteristics of what is needed for creating such a solution. A complete list of the QUEST tenets can be found at [26]; however, the current research is focused on the tenets for phenomenal states (i.e., qualia), which are shown below.

1. **Subjective Aspects of Qualia:** Qualia are the subjective qualities evoked by a stimulus. They only exist in the mind of the animal sensing the stimuli.
2. **Not Derivable:** The qualia are not derivable from the stimulus by any other animal. Qualia are so distinct that the same stimulus presented at a different time to the same animal could evoke a different quale.
3. **Not Measurable:** Qualia are not measurable by any external agent. There is no set of measurements that can be taken to explain what it is like for an animal to experience a specific quale.
4. **Qualia Spookiness:** The fact that qualia are only accessible by the animal that generates them make them ‘spooky.’ We call this gap between the externally observable stimuli and the only internally accessible qualia the S-Q gap.

5. **Abstract Qualia:** Abstract is defined by philosophers as 'not being reducible by sensor transduction.' All qualia are abstract concepts.
6. **Internally Generated:** Qualia are evoked as a result of stimulation. That stimulation can be the result of sensing or can be internally generated, e.g., by thinking or dreaming.
7. **Processes that act on Qualia:** There exists a set of processes that manipulate the internal qualia representation. These processes generate efficient representations such as the formation of hierarchies to generate compound qualia.
8. **Evolving Qualia:** The qualia-based representation facilitates anticipating, detecting, distinguishing, and characterizing entities. That representation can change and be manipulated.
9. **One Quale at a Time:** It is only possible for one quale to be experienced at a time. Multiple solutions compete to be experienced.

10. **Qualia Theory of Relativity:** Qualia-based representations build a world model that is completely relative. Each individual quale can only be characterized relative to other qualia.

11. **Negative Aha:** QUEST must not only be able to identify what it knows, but also what it doesn't know. This is termed 'the known unknown.'

12. **Qualia Sensors – Measurement Units:** It is not yet clear that the conventional approach of mapping sensory measurements immediately to numbers doesn't lead

us down a path where we can never get to a qualia-based representation. QUEST may involve a new approach to non-numeric sensors.

13. **Intent – Theory of Mind (ToM):** Theory of mind is the act of computing the quale of ‘mindness’ by an animal. It is one of the most important links for the quale of self.

14. **Self:** The concept of self involves being able to distinguish in the world model what is under one’s own control and what is external. This computation will arise from interaction with the environment.

15. **Chinese Room and Zombies:** It is impossible to engineer a system that can give intelligent responses to arbitrary queries without having any understanding of the queries themselves.

Although there is no standard design methodology for developing a general purpose MC framework for investigating phenomenal states, several researchers have provided guiding principles for different aspects of this problem, as described above. That is, Crick and Koch recommended attributes for developing a conscious framework, Morsella provided a list of functional tenets (i.e., SIT) for phenomenal states, and proposed characteristics/representations of qualia (i.e., phenomenal states) were given by Rogers et al.

This research uses a subset of the conscious framework guiding principles to serve as design criteria for the proposed architecture. The subset of guiding principles selected to design CASE are only those appropriate for MC systems and not just applicable for human consciousness (e.g., penumbra and meaning). Furthermore, a second filter was used for only selecting the principles that are useful for inference and reasoning since they are key features currently absent in SHM systems. The selected framework guiding principles used to design CASE are listed below.

1. Unconsciousness
2. Consciousness
3. Coalition of neurons
4. Explicit representation
5. Attention and binding

The assumption is made that designing CASE using the selected framework principles to mimic characteristics of consciousness, will also elicit the associated functions and attributes of phenomenal states. Nevertheless, using these guiding principles does not ensure the generation of phenomenal states. However, by using the recommended guidance, a framework emerges to further the investigation of MC and phenomenal states.

2.6 Summary

This chapter reviews the most cited theories of consciousness and their investigation in MC systems. Additionally, it discusses the characteristics of consciousness from a functional and phenomenal perspective. Proposed tenets for these characteristics were provided along with suggested attributes for a conscious framework.

III. Methodology

3.1 Overview

In this chapter, the proposed architecture is described in detail. Furthermore, each of the selected conscious framework attributes is illustrated. Lastly, the corresponding SIT functions and QUEST tenets of the selected attributes are identified.

3.2 Conscious Architecture for State Exploitation (CASE)

CASE is a general purpose architecture that autonomously generates state information while situated in some environment to enhance decision making. The CASE computational philosophy is inspired by the cognitive information processing of humans. The CASE framework is designed to mimic the integration of low-level and high-level cognitive functions. Specifically, CASE incorporates specific characteristics of the unconscious and conscious processes of human cognition. Figure 4 illustrates CASE, consisting of two integrated systems: a perceptual system (unconscious process) and a conceptual system (conscious process) [11].

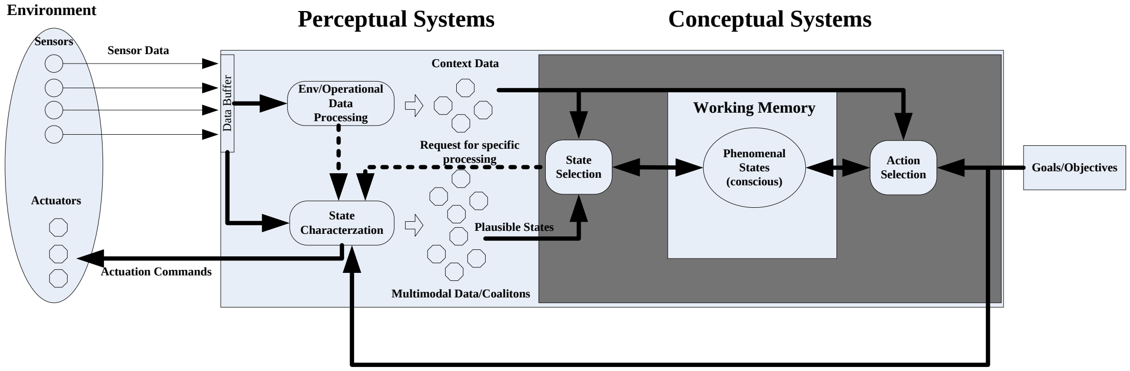


Figure 4. Conscious Architecture for State Exploitation (CASE)

3.2.1 Perceptual System

The perceptual system processes sensory data acquired from the environment to quickly compute state estimations (reflexive) via pattern recognition techniques. Data sensed from the environment are capable of being processed in parallel from different sensing modalities (e.g., temperature, state estimates, etc.). Furthermore, sensing modalities can be combined (e.g., form coalition) to increase the reliability of state estimations. The environment can also be influenced by request of the perceptual system through implementation of selected actions performed by actuators (see Figure 4). The output of the perceptual systems produces one or more state estimates (i.e., plausible states) that compete to enter the conceptual system (i.e., global workspace) for further processing.

3.2.2 Conceptual System

The conceptual system comprises long-term and working (i.e., short-term) memories. Long-term memory stores procedural, semantic, and episodic knowledge regarding the environment and application-specific information. Procedural memory provides knowledge related to particular action rules needed to achieve a given goal. General knowledge about the environment is stored in sematic memory, and episodic memory contains information regarding past experiences (i.e., past selected states). CASE uses working memory for reasoning and deliberating over state estimations. It contains all the relevant information pertaining to the current situation such as state estimations, goals and action rules. If additional information is required, working memory can also query data from long-term memory and the perceptual systems to aid in the decision making process.

3.3 Functions and Processes

The perceptual system consists of two main modules: *environmental/operational data processing* and *state characterization* (low-level processes). The conceptual system also consists of two main modules: *state selection* and *action selection* (high-level processes). Each of these main elements is described below, and examples related to an assumed SHM application are provided.

3.3.1 Environmental/Operational Data Processing

The *environmental/operational processing* module acquires measurement data from the environment via sensors to provide context data regarding the manner in which the observed system (e.g., airframe, automobile, etc.) is being operated and the environment in which the system is operated. For example, these data could include the external temperature of the surrounding environment and/or the operational speed of the system being monitored (see Figure 5). Data collected by this module are primarily based on first principle sensing, which provides physics state information. For example, temperatures are environmental data, and load levels and load cycles are operational data. Both context data types are used within the *state selection* module for computing the anticipated states via physics-based models (i.e., simulation). Additionally, context data could be integrated with the *state characterization* data for enhancing state estimations.

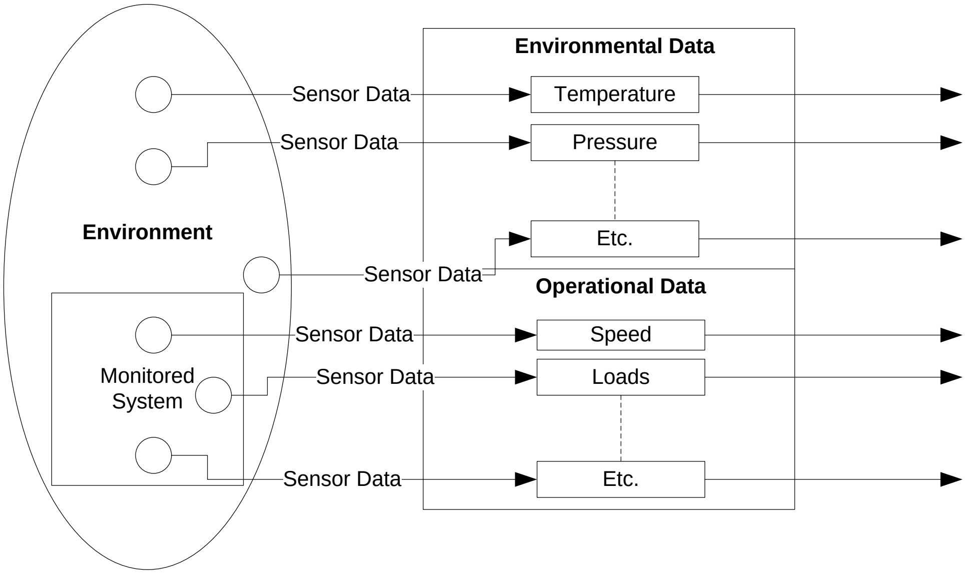


Figure 5. Environmental/Operational Data Processing Module

3.3.2 State Characterization

State characterization is the process used to estimate health states via sensors and pattern recognition methods. In Figure 6, sensor data from the monitored system are processed using damage estimation techniques. These state estimations could be performed using several combinations of sensing modalities and algorithms. For example, all damage algorithms could process data differently, using different sensors of the same modality for providing an estimate of the same state. Likewise, all processing algorithms could estimate different states using the same sensors but different modalities.

The dashed line entering the top of the *state characterization* module indicates that environmental/operational data could be used to supply context data. For example, temperature compensation could be applied to sensor data based on thermocouple readings (see Figure 6). That is, the damage algorithm could be adapted to compensate for the effects caused by the operational and environmental conditions. The output of the state characterization stage is one or more state estimates (i.e., plausible states) of the observed system. These outputs could also be combined (i.e., form coalitions) to improve state estimations.

There are numerous ways the *state characterization* module could be configured for acquiring specific data and computing state estimations (e.g., sensing modalities, data processing methods, etc.). For example, this module could perform specific interactions with the environment via actuators for acquiring specific data. Additionally, the *state characterization* module could be reconfigured to process data in a specific way to enhance state estimations. Both of these features could occur in real-time at the request of the perceptual systems via the *state section* module. At this stage, state estimates give information related to presence of damage and the degree of damage, such as crack length estimates for a monitored location on an aircraft.

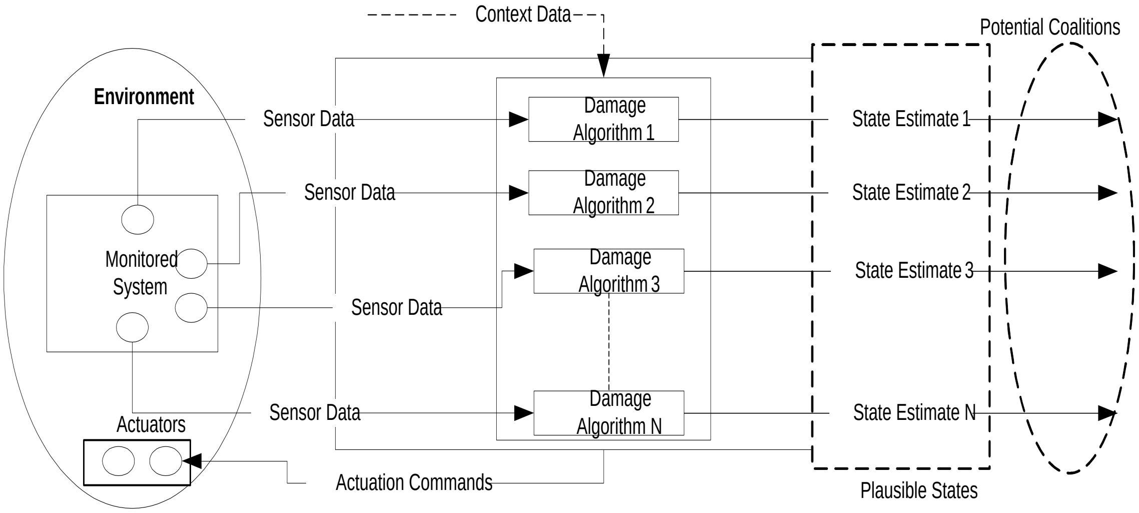


Figure 6. State Characterization Module

3.3.3 State Selection

State selection involves using context data and/or physics-based models (i.e., simulation) to refine state estimates. Context data are provided by the *environmental/operational processing* module to perform simulations using a physics-based model. For example, load data can be used in this simulation to predict the existence of a crack, given knowledge of the material properties and geometry of the part. These predictions are combined with the current state estimates (one or more) from the *state characterization* module and the past selected states to determine if they are logical and feasible (i.e., do not violate the laws of physics). This consistency and reasonability assessment is performed using a state selection algorithm. Estimates that fail this test are deemed invalid and do not further influence the selected answer. The selected answer could be any one or a combination of the state estimates. Furthermore, it could be a state estimate computed by the simulation. The objective of the *state selection* module is to generate stable and consistent state estimates appropriate for actionable decision making and not necessarily a precise state assessment. Figure 7 depicts the state selection process.

Requests for specific data and processing can be made to support the state selection procedure, as indicated by the dashed line, to the *state characterization* module (see Figure 4). The specific processing may be as simple as requesting repeated measurements from the same sensors, or even computing state estimates from different sensors using secondary methods that are possibly more computationally demanding. Once a decision is reached, the selected state is used within the *action selection* module.

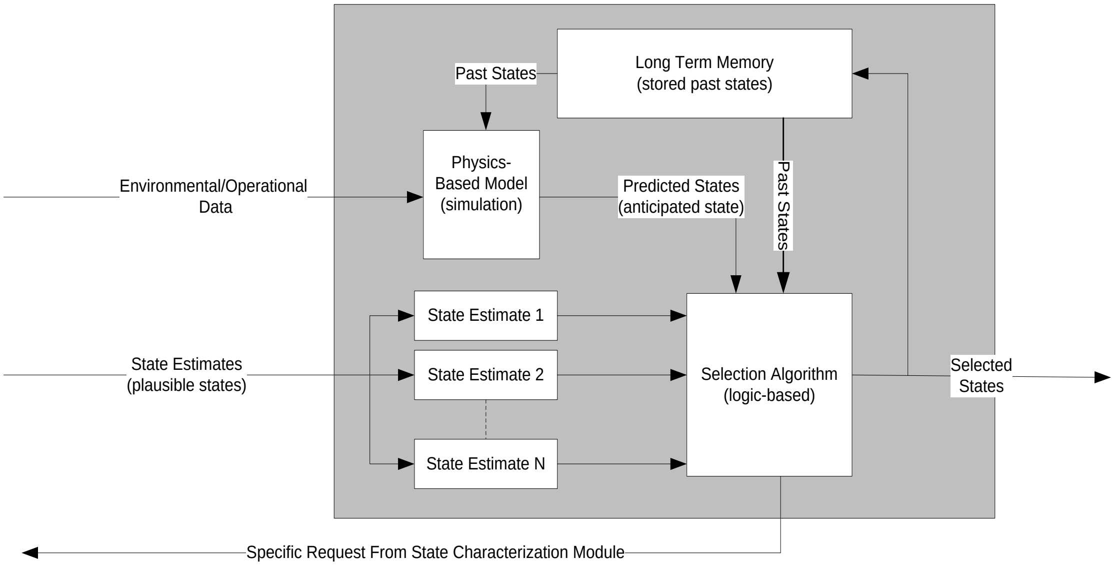


Figure 7. State Selection Module

3.3.4 Action Selection

The *action section* module combines context data (i.e., environmental/operational data), state selection data, and goal-based data to select the most appropriate action for achieving the current objective. This module uses selected states, anticipated environmental/operational data and goals/objectives data as input into a simulation (i.e., physics-based model). This simulation is performed to identify potential problems that could occur as a result of a particular action. If the simulation anticipates major issues with performing a specific action, then an alternative action with fewer concerns is recommended.

The goal-oriented actions will differ based on the needs of each individual application. For example, a maintainer may establish rules to be notified when a visual damage inspection is needed. Or, a mission commander might design rules that report risk of component or mission failure for the given current structural state and anticipated flight profile. The action selection module of CASE is shown in Figure 8.

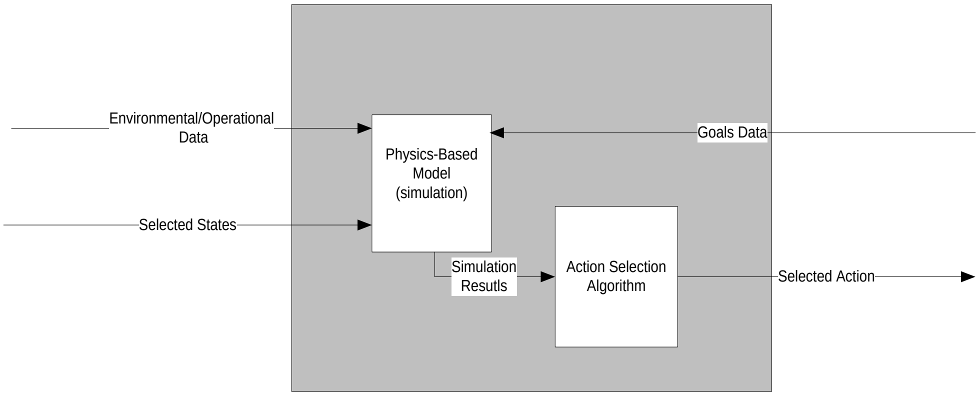

Environmental/Operational Data
Selected States
Physics-Based Model (simulation)
Goals Data
Simulation Resutls
Action Selection Algorithm
Selected Action

Figure 8. Action Selection Module

3.4 CASE Operational Characteristics

The operational features of CASE were designed to include selected attributes of a conscious framework (Chapter 2). These features were chosen based on their applicability to a MC system and the demonstrated application (i.e., SHM). Furthermore, the corresponding functionality of SIT was also incorporated. Explicitly, CASE was developed using the guiding principles selected in Chapter 2. Using the functional modules of CASE described above, these guiding principles are illustrated via Functional Flow Block Diagrams (FFBD) for the selected conscious framework attributes.

3.4.1 Unconscious

The unconscious attribute is a mode in which actions in response to sensory inputs are rapid, transient and stereotyped. Unconscious attributes could also be thought of as automatic or reflexive responses analogous to a “knee jerk” reaction. Additionally, the primary flow of information is most likely feed-forward and bottom-up (i.e., from left to right in the diagram below). Figure 9 illustrates the unconscious mode in CASE.

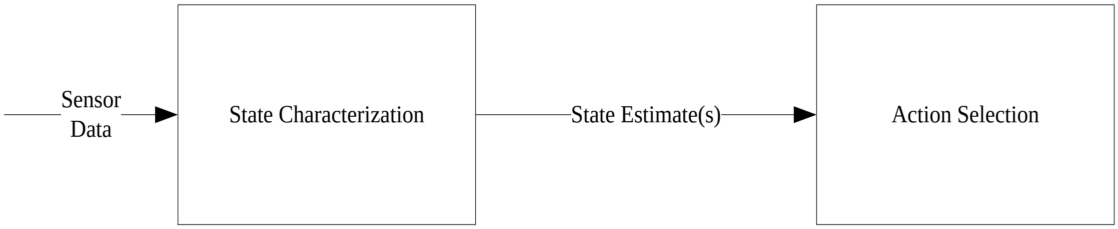


Figure 9. Unconscious Mode of CASE

In Figure 9, sensory data from the environment enter the *state characterization* module in CASE. This module computes one or multiple state estimates. In this mode, the *state characterization* module must yield one output (i.e., state estimate) in order to provide actionable data to the *action selection* module. If multiple state estimates are computed, they must be combined (e.g., form a coalition) in order to meet the "one output" requirement. Although the depiction of the unconscious process in Figure 9 does not make use of any environmental/operational data, these data could be used to aid in the state estimation process. The computed state estimated then enters the *action selection* module, where the corresponding action is immediately performed without deliberation or reasoning (e.g., state selection module).

In the unconscious mode, no data modulate the phenomenal field since it is a reflexive response. This supports the SIT theory that *relatively few kinds of information require conscious interaction, because many kinds of information can interact unconsciously* [22].

3.4.2 Consciousness

Contrary to the unconscious (i.e., zombie modes) process, consciousness is a more thoughtful and slower process in CASE. It uses reasoning and deliberation to determine the most appropriate response to a given sensory input. When used together with the unconscious system (i.e., zombie system or perceptual system), the conscious system may interfere (e.g., veto, change, etc.) with the unconscious system's reflexive response to ensure that the most logical action is taken. This process is evoked when the unconscious system response is inconsistent, unstable or illogical. In the conscious mode, data flow in both directions (i.e., bottom-up and top-down). The conscious mode of CASE operation is depicted in Figure 10.

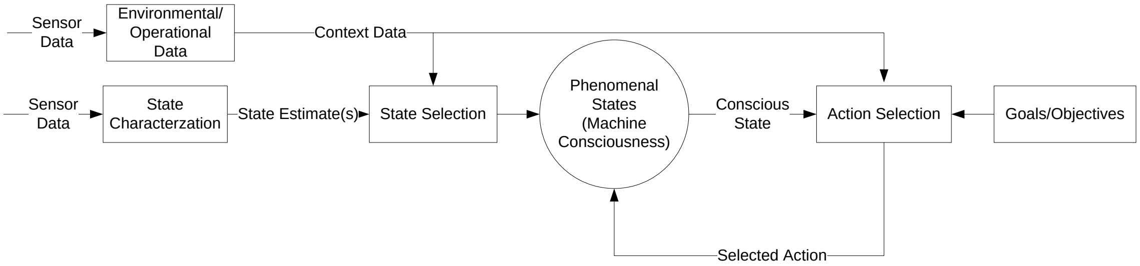


Figure 10. Conscious Mode of CASE

Figure 10 shows sensory data from the environment entering the *state characterization* and *environmental/operational processing* modules. Similar to the unconscious mode, the *state characterization* module can compute one or more state estimates (e.g., damage size). However, in the conscious mode, the *state characterization* module is permitted to output multiple state estimates since a state selection process is performed next. These multiple state estimates at this stage can be referred to as plausible states. The *environmental/operational processing* module computes context data related to the manner in which the observed system is being operated (e.g., stress levels, velocity, etc.) and in what type of conditions (e.g., high temperature, low temperature).

The *state selection* module combines context data and state estimates for reasoning to determine if the plausible states (i.e., estimated states) are logical and feasible (i.e., do not violate the laws of physics). Results that fail the reasoning process are deemed invalid and do not further influence the selected answer (i.e., lose the competition and do no enter the global workspace). Conversely, the valid results allow entrance to working memory (i.e., global workspace) and can become the focus of attention (i.e., artificially conscious) if they are selected (i.e., enter the "spotlight"). Once an artificially conscious state is realized (i.e., selected), it is then used during a deliberation process within the *action selection* module. This module integrates context data, selected state data and goals/objectives data for deliberation to ensure an appropriate action is taken given the current situation. The selected action also becomes artificially conscious once the decision is made.

Operations in the conscious mode employ three of the SIT functional tenets and two QUEST tenets. The SIT tenet *interaction across systems (i.e., modules) requires conscious processing* is used [22]. A SIT diagram of CASE is shown in Figure 11. Notice that three modules are combined in the phenomenal field before an action is performed. This is required in order to select the best plan of action for a given situation as discussed in the *state section* and *action section* modules described above. Additionally, the conscious mode supports the SIT tenet of *outputs of different response systems requires phenomenal states to interact* [22]. The conscious mode in CASE uses the phenomenal field twice for interaction before an action is determined. First, it is used in the *state selection* module for selecting from plausible states. This process integrates data from the *environmental/operational processing* and *state characterization* modules before a selection is made using the process discussed earlier (see Figure 7). Second, the *action selection* module uses data from *environmental/operational processing*, *state characterization* and *goals/objectives* modules as described above (see Figure 8). This process also makes use of the phenomenal field (i.e., conscious representation) before an action is executed. Since the *state selection* and *action selection* modules both integrate data from different systems before responding, it becomes obvious that the SIT principle of *response systems incessantly modulate the phenomenal field* is demonstrated in the conscious mode [22]. That is, anytime a module's output becomes artificially conscious, it modulates the phenomenal field. See Figure 11.

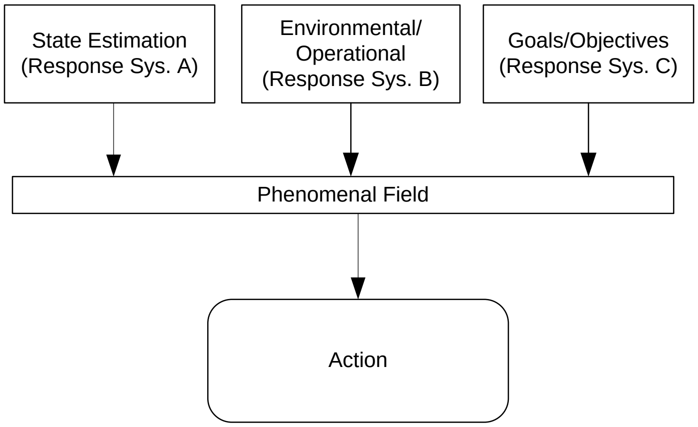

State Estimation (Response Sys. A)
Environmental/ Operational (Response Sys. B)
Goals/Objectives (Response Sys. C)
Phenomenal Field
Action

Figure 11. Conscious Mode Phenomenal Field Modulation [22]

The CASE operation in the conscious mode utilizes the QUEST tenet *"not measurable"* [28]. This tenet can be described as a process in which the unconscious system (i.e., sensory system) results are changed, manipulated or replaced by the conscious system to generate a logical and thoughtful response. CASE also uses the QUEST tenet *"evolving qualia"* in the conscious mode [28]. During the *state selection* and *action selection* processes, CASE uses a simulation (i.e., representation) that is manipulated via past and current state data. This simulation facilitates the detection, characterization, distinction, and anticipation of current and future state estimations. A detailed example of these processes will be provided in Chapter 4.

3.4.3 Coalitions of Neurons

Coalitions of neurons take place at two stages within CASE. During the state characterization process, neurons (i.e., features) could form coalitions to generate state estimations. That is, each output from the *state characterization* module could comprise coalitions of features. This condition could occur within a single sensing modality or across sensing modalities. Similarly, the output of the *state characterization* module could be combined to form coalitions before entering the state selection module. Figure 12 illustrates this process.

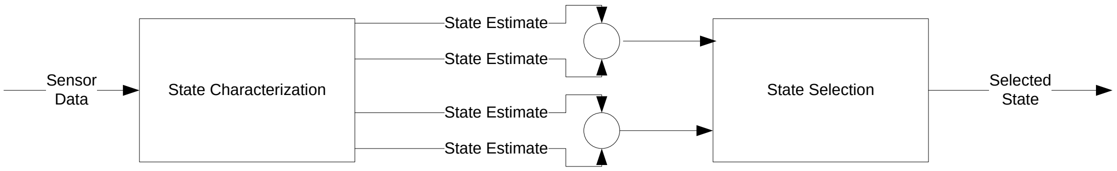


Figure 12. Coalitions of Neurons in CASE

The SIT function that the coalitions of neurons displays is *outputs of different response systems requires phenomenal states to interact* [22]. Since the coalition of neurons could be from different modalities, each modality could be considered a different response system. For example, imagine trying to recognize an object in the environment via sensory observation. If more than one of the senses (e.g., seeing and hearing) is used to characterize the object, the SIT principle described above is exercised. This is analogous of what happens within CASE. In the given example, this coalition process may or may not modulate the phenomenal field depending on whether it was performed in a conscious or unconscious mode.

3.4.4 Explicit Representations

The explicit representations characteristic in CASE occur in the *state characterization* module. Each sensory processing element within the *state characterization* module has an explicit representation. This representation allows each sensory element to respond to specific features reflexively without further processing. For example, one explicit representation might be able to detect structural damage. Yet another explicit representation could have the ability to identify damage locations. Both have the innate ability to respond autonomously via their explicit representation without additional processing. Figure 13 depicts this process.

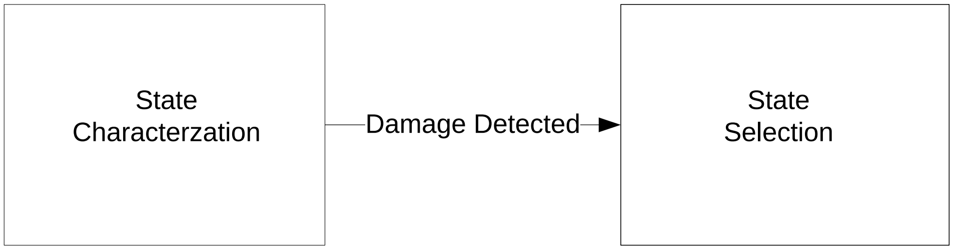


Figure 13. Explicit Representations in CASE

In Figure 13, assume that the *state characterization* module has an explicit representation to detect damage. The damage detection data are sent directly to the *state selection* module, where it competes for access to working memory (i.e., global workspace). Since damage detection is an explicit representation and is computed in a single module, no SIT functions are illustrated during this process.

3.4.5 *Attention and Binding*

Attention is divided into two forms. It can be rapid, saliency-driven and bottom-up or slower, volitionally controlled and top-down. In CASE, the bottom-up attention method is performed via the state characterization process. Consider Figure 14. Assume that an anomaly is detected in the sensor data via the *state characterization* module. This abnormality gets attended to via the state selection process. Furthermore, the appropriate action is taken during the action selection process to ensure the goals/objectives are achieved in spite of this anomaly.

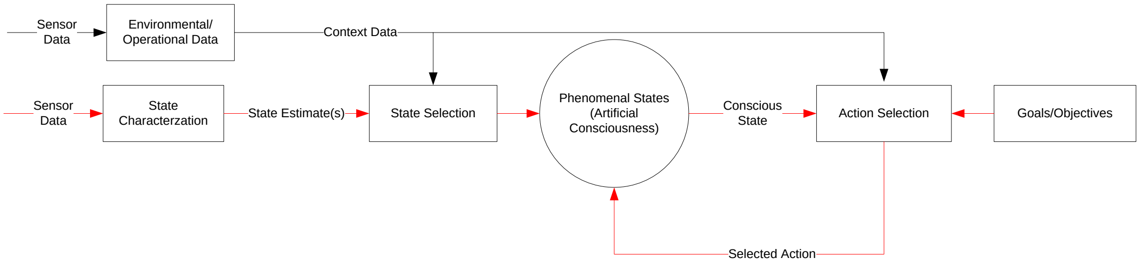


Figure 14. Bottom-up Attention Process in CASE

The top-down attention is driven by the *goals/objectives* and executed using the *action selection* module. For instance, the *action selection* module can focus the attention of the system (i.e., MC system) by requesting specific data from the *state characterization* module to aid in the decision making process. In Figure 15, assume the *goals/objectives* module makes a request for specific sensor data. This request is sent to the *action section* module; in turn the *action section* module commands the *state* *characterization* module to acquire the requested data. These data are then processed in a feed-forward manner to become artificially conscious. This process will continue until an acceptable solution is reached.

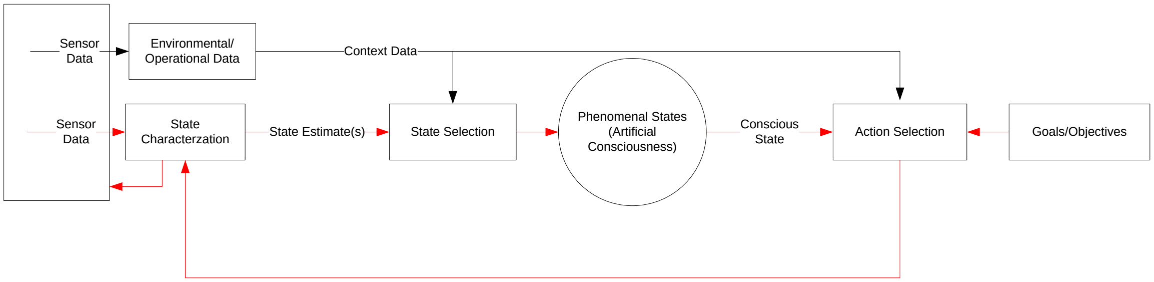


Figure 15. Top-down Attention Process in CASE

Binding is a process that brings together different aspects of an object or event. The philosophy behind binding is similar to an investigator interviewing two witnesses at the scene of a car accident who observed the same incident from different angles. The goal is to collect all relevant aspects of the data for generating a complete representation for making the most accurate decision possible. CASE performs binding by integrating data from sensors physically positioned in different locations. These data could be of the same type or of different modalities. In addition, these data could be acquired with high or low resolution. Nevertheless, these data sets are combined to corroborate each other for providing the best possible response with the available data.

The SIT tenets exhibited in the attention and binding attribute depend on the way the data are processed. For example, if the attention/binding processes only need data from a single module with no interactions, then the SIT functions are not exhibited. Conversely, if the attention/binding processes require data interactions from different modules, then one or more SIT functions will be present. The attention process in CASE exhibits the QUEST tenet *one quale at a time* [28]. During the attention process in CASE, whether bottom-up or top-down, one state is attended to at a time. This is true during the *state selection* and *action selection* processes, which both use artificial consciousness for decision making.

Table 1 summarizes the guiding principles illustrated in CASE for this investigation. CASE was designed, developed and demonstrated in an SHM application incorporating the philosophies listed in Table 1. Details of this design and the demonstration results are discussed in Chapter 4.

Table 1. Guiding Principles Demonstrated in CASE

Framework Attributes	***SIT Tenets***	***QUEST Tenets***
Unconscious	-Relatively few kinds of information require conscious interaction, because many kinds of information can interact unconsciously.	
Consciousness	-Interaction across systems (i.e., modules) requires conscious processing. -Outputs of different response systems require phenomenal states to interact. -Response systems incessantly modulate the phenomenal field.	-Not Measurable -Evolving Qualia
Explicit Representation		
Coalition of Neurons	-Outputs of different response systems require phenomenal states to interact.	
Attention/ Binding	-Depends on processing methods used.	-One Quale at a Time

3.5 Summary

This chapter provides a general overview of the CASE architecture. In addition, the four main modules of CASE are described in detail. Lastly, the selected characteristics of a conscious framework are illustrated via CASE, and the associated SIT/Quest tenets are discussed.

IV. Demonstrated Application

4.1 Overview

This chapter describes the demonstration of CASE in an structural health monitoring (SHM) application. First, the current method for ensuring structural integrity of USAF airframes is briefly discussed. Next, an overview of SHM is given, followed by the experimental setup. Finally, experimental results are presented comparing CASE with the current practice for maintaining aircraft structures.

4.2 Aircraft Structural Integrity Program (ASIP)

The United States Air Force (USAF) utilizes the Aircraft Structural Integrity Program (ASIP) to service and maintain its airframes. The goal of ASIP is to ensure the desired level of structural safety, durability, and supportability with the least possible economic burden throughout the aircraft design service life [9]. USAF aircraft structures are currently designed using a “damage tolerant” philosophy, wherein structures are designed to retain the required residual strength for a period of unrepaired usage after the structure has sustained specific levels of fatigue, corrosion, accidental, and/or discrete source damage [9]. ASIP currently manages damage using a schedule-based maintenance philosophy by establishing predefined maintenance intervals for performing manual inspections. This approach requires vehicles to be removed from service at predetermined times regardless of their actual condition. In most cases, inspections performed during this process do not find any damage, and the airframe is returned to service until the next inspection interval. The schedule-based maintenance approach works well for ensuring structural integrity. However, it is very costly, labor-intensive, and reduces aircraft availability. Furthermore, Operational and Support (O&S) costs are continuously rising due to the frequent inspections required to maintain aircraft safety in aging fleets [17].

4.3 Structural Health Monitoring (SHM)

SHM can be defined as automated methods for determining adverse changes in the integrity of mechanical systems [12]. The need for and benefits of SHM systems for civil, military, and aerospace applications have been documented by many researchers [14] [20] [24] [32]. The ultimate goal of SHM is to provide an automated and real-time assessment of a structure's ability to serve its intended purpose. Structural health assessments consist of a diagnosis and prognosis of the monitored structure. The diagnosis should include the detection, localization, and assessment of any damage, while the prognosis provides information regarding the consequences of the diagnosed damage. The prognosis might be that the structure is as good as new, safe to operate for only a certain number of flight hours, or that immediate repair is required. Knowledge regarding the state of the structure is increased with each level of abstraction.

SHM systems are typically comprised of in-situ or embedded sensors and processing algorithms. The algorithms are used to interpret sensor data to discriminate between different damage states in order to provide an accurate damage assessment and corresponding prognosis. Various processing steps may be performed by the SHM system to transform the data into different forms that enhance the damage assessment ability. Most SHM systems process sensory data using pattern recognition methods to classify structural states [19]. Development of SHM systems based on pattern recognition requires training data from all anticipated damage states and operational environments to be effective. The training data are used to design a classifier, and the resulting performance is evaluated by scoring the classification results from data not utilized during the design or training phases.

The integrity of a structure can be described at different levels of detail. The two fundamental levels, detection of damage and its location, are each useful in their own rights. Using only results from the damage detection and localization levels, inspection time and costs could be reduced. Most SHM investigations have focused on developing quick state assessments (i.e., reflexive techniques) for achieving the fundamental levels of SHM. These reflexive approaches have achieved limited success for damage detection and localization [13]. However, information provided by the two higher levels of SHM, relating to quantifying the degree of damage and ultimately an assessment of the consequences of damage in terms that are meaningful to maintainers, operators, and commanders, could lead to further improvements in operation. Figure 16 illustrates a notional SHM system (high and low levels). Exploiting the full operational benefits of SHM requires a new methodology for information processing. SHM is a well-suited application for demonstrating CASE.

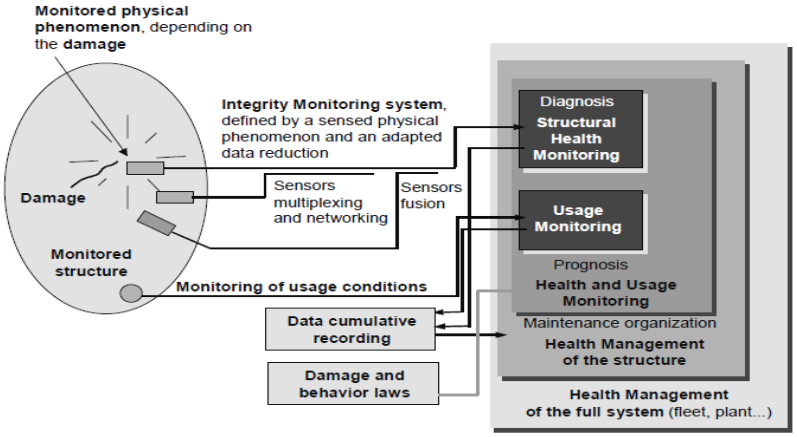

Monitored physical phenomenon, depending on the damage
Integrity Monitoring system, defined by a sensed physical phenomenon and an adapted data reduction
Damage
Sensors multiplexing and networking
Sensors fusion
Monitored structure
Monitoring of usage conditions
Data cumulative recording
Damage and behavior laws
Diagnosis
Structural Health Monitoring
Usage Monitoring
Prognosis
Health and Usage Monitoring
Maintenance organization
Health Management of the structure
Health Management of the full system (fleet, plant...)

Figure 16. Notional SHM System [5]

4.4 Experimental Setup

CASE was used to autonomously monitor the integrity of a flight critical airframe component, a representative wing spar attachment lug, under simulated flight loads. As the load cycles accumulated and the airframe component began to fracture, CASE computed an output recommending a maintenance action be performed by a maintainer. The results were compared and contrasted with ASIP using DoD metrics [10].

The test article used for this investigation is a representative single wing spar assembly made of 6061-T6511 extruded aluminum that was subjected to flight-like fatigue loading [15]. Although 2024 and 7075 are the most common alloys used in aircraft, 6061 was selected for this experiment because it is less expensive and readily available. One end of the spar was mounted to a test fixture representing the wing attachment to the fuselage. The opposite end of the spar was loaded in fatigue using a hydraulic actuator to emulate wing deflection during flight. The test configuration is shown in Figure 17.

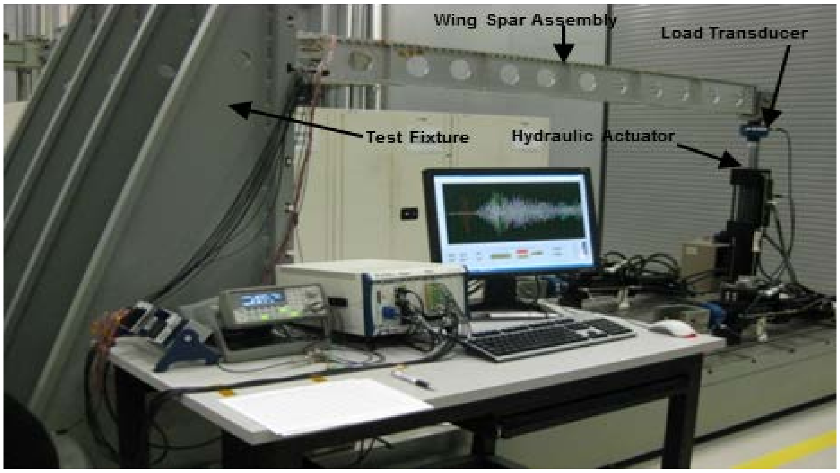


Figure 17. Experimental Setup

A finite element analysis was performed on the test article to determine the critical locations that require monitoring. It was determined that the wing spar attachment lug was most likely to fracture first under fatigue loading. During cyclic loading, corner cracks were predicted to initiate at the shoulders of the lug and grow horizontally (A-direction) and vertically (C-direction), as shown in Figure 18.

The commercially available AFGROW (Air Force Growth), which is a physics-based fracture mechanics software, was used to provide estimates of crack initiation and growth. The loading profiles were assumed to be sinusoidal with constant peak load amplitude of 1,000 lbf and a minimum load of zero. Under these conditions, the critical crack lengths (predicted failure size) in the horizontal and vertical directions were found to be 0.35” and 0.70”, respectively. Crack initiation estimates can be approximated from fatigue testing performed on un-notched (pristine) 6061-T6 specimens [15]. For this experiment, an assumed initiation crack size of 0.02” was used. Therefore, the estimated time for a 0.02” crack to initiate was determined to be 10,000 cycles. The fatigue life of the lug was estimated using AFGROW as well. Assuming an initial flaw size of 0.02”, under a constant peak applied loading of 1,000 lbf, the lug was predicted to fail at 14,500 cycles. Hence, the estimated lifecycle of the lug under the test conditions was estimated to be 24,500 cycles. Figure 19 shows the results of the crack initiation and growth predictions for selected loading conditions.

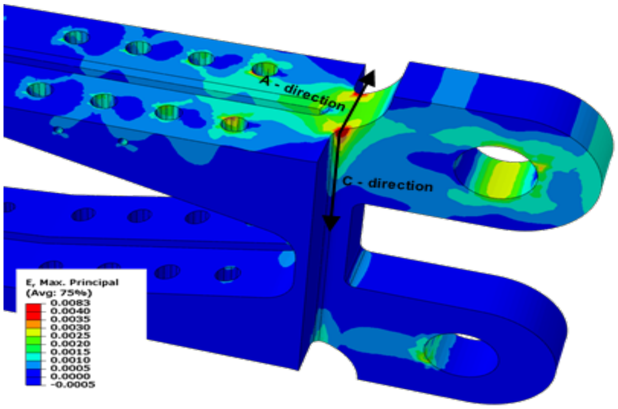


Figure 18. Model of Wing Spar Attachment Lug

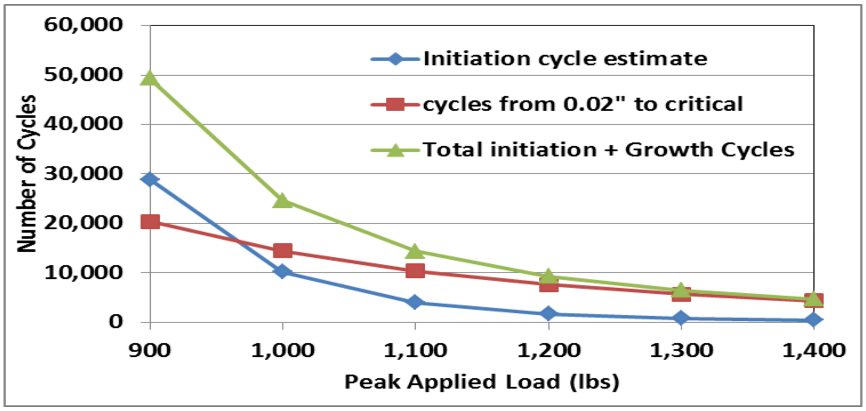


Figure 19. Crack Initiation and Growth Predictions

Throughout the laboratory fatigue testing, measurements of the visual crack size (i.e., truth data) and SHM data were collected during pauses in the fatigue cycling.

Visual crack size measurements were performed using Florescent Dye Penetrant, and SHM data were generated using piezoelectric transducers (sensors and actuators) bonded to the lug [20]. The interval between data collections was based on the measured visual crack size. Measurements were made every 1,000 cycles until a crack was visually detected. After visual detection, measurements were made every 500 cycles until the longest observed crack reached 0.30”. Once the longest crack reached 0.30”, measurements were made every 250 cycles. The experiment was terminated when the longest crack reached 0.70”. This schedule provided 123 measurements over 70,000 fatigue cycles for use in the simulation of the ASIP and CASE processes.

4.5 Demonstration Results

4.5.1 ASIP

Because of the conservative nature of ASIP, it is assumed that all critical airframe components have an initial flaw size to account for any damages that could have occurred during the manufacturing and maintenance processes. Generally, ASIP assumes a 0.05” flaw because it is equivalent to the minimum detectable flaw size of a typical structural inspection. For this reason, a 0.05” flaw was assumed to exist in the lug component. Using AFGROW with the loading profiles used for this testing, the estimated fatigue life, or the time required for an initial crack of 0.05” to grow to the critical crack length for the lug, was approximately 8,615 cycles. The ASIP process usually establishes inspection intervals by performing the first manual inspection at half the estimated fatigue life, and the next inspection at the estimated fatigue life. Therefore, the ASIP inspection interval used for this experiment was roughly 4,300 fatigue cycles (8,615 cycles / 2).

During testing, cracks initiated from both shoulders (left and right sides) of the lug and propagated in both the horizontal and vertical directions as expected. The first noticeable cracks were at 43,000 cycles in the vertical direction, with lengths of 0.091" and 0.08" on the left and right shoulders, respectively. Cracks in the horizontal direction were not detected until 47,500 cycles with sizes of 0.058" and 0.048", respectively. As noted above, the estimated times for a 0.02" crack to initiate was 10,000 cycles. Because the inspection technique used for this experiment could only detect flaws above 0.05", the 0.02" crack initiation assumption could not be verified. However, it is still interesting to compare the estimated and measured cycles required for crack initiation and growth as shown in Table 2.

Table 2. Error in Crack Initiation and Growth Predictions

Crack Direction and Side	**Cycles for Crack Initiation**			**Cycles for Crack Growth to Critical**		
	Estimated	Measured in (mm)	Percent Error (%)	Estimated	Measured in (mm)	Percent Error (%)
A-dir left side	17,000	47,500 0.058 (1.47)	179	7,500	20,250 0.360 (9.14)	170
A-dir right side	16,000	47,500 0.048 (1.22)	197	8,500	17,750 0.360 (9.14)	109
C-dir left side	18,500	43,000 0.091 (2.31)	132	5,500	27,000 0.520 (13.21)	391
C-dir right side	18,000	43,000 0.080 (2.03)	139	6,500	27,000 0.680 (17.27)	315

In Table 2, the error in predicting crack initiation cycles was between 132% and 197% for all of the cracks. Table 2 also lists the estimated and measured cycles for the cracks to grow from the initial crack size measured to the critical crack length. The crack propagation errors range by almost a factor of four, from 109% to 391%. These ranges are typical of crack propagation behavior, as it is not uncommon for fatigue crack growth predictions to vary by a factor of four [33].

Since the lug is a fracture critical component, ASIP would require periodic inspections to ensure fatigue cracks do not initiate and grow beyond the critical crack length before being repaired. Using the ASIP-established inspection interval of every 4,300 cycles for this component, the lug would be inspected approximately ten (43,000/4,300) times before any damage is detected for cracks in the vertical direction and eleven (47,500/4,300) times for cracks in the horizontal direction. These inspections in which no damage is detected are significant since the cost for inspecting similar components on fielded aircraft range from approximately $1,000 to $120,000 per inspection based on various factors (e.g., aircraft configuration, type of inspection, coating removal and restoration, etc.) [23].

4.5.2 CASE

CASE was applied to the same representative aircraft component. The shoulder regions of the lug were instrumented using bonded Kapton-encapsulated piezoelectric transducers (PZTs) as shown in Figure 20. During the SHM data collections, ultrasonic elastic waves were transmitted through the shoulder regions of the lug, from a rectangular actuation PZT to six sensing PZT disks for each side (i.e., left and right sides). The actuation signals were 5½ cycle windowed tone bursts with center frequencies ranging from 400 kHz to 1 MHz in 100 kHz increments. Sensor data was recorded with a 10 MHz sample rate and 12-bit amplitude resolution. Load data was also collected throughout the demonstration via a force transducer attached to the tip of the hydraulic actuator. Figure 21 depicts the instantiation of CASE as demonstrated. Sections 4.5.2.1 through 4.5.2.4 will describe how the CASE modules processed the acquired data.

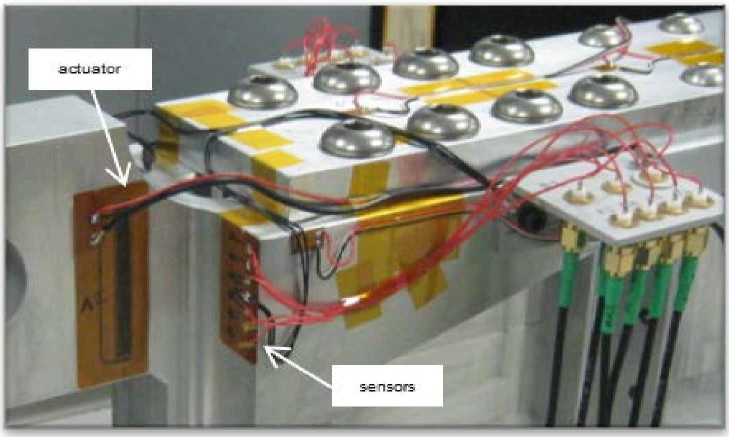


Figure 20. PZT Sensors/Actuator Installed on the Wing Spar Attachment Lug

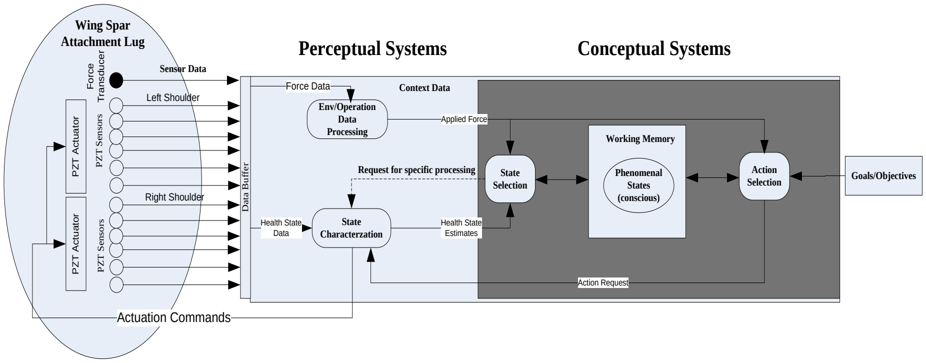


Figure 21. CASE as Demonstrated

4.5.2.1 Environmental/Operational Processing

For this demonstration, the environmental/operational data processing module computed loading and cycle count information via the load transducer attached to the hydraulic actuator (see Figure 17). During cyclic loading, the actual applied loads were measured and acquired throughout the experiment. These data provided operational information regarding the actual loading profile experienced by the wing spar assembly and the corresponding duration or number of cycles. Figure 22 depicts a block diagram of this process. This information serves as context data for the *state selection* and *action selection* modules.

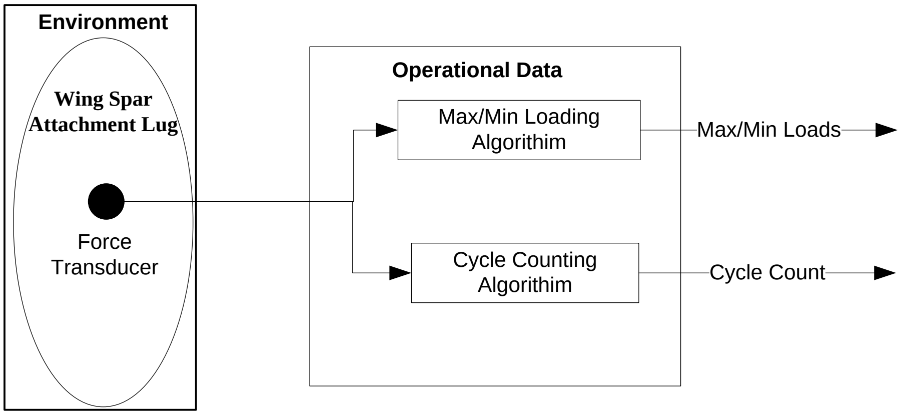


Figure 22. Environmental/Operational Module of CASE as Demonstrated

4.5.2.2 State Characterization

The structural state characterized during this demonstration was crack size. The fundamental feature for crack size estimation is based on a damage index derived from the correlation coefficient between a reference and test measurements. The reference measurements were taken at cycle 1,000. This was done to give the test article, sensors, etc., time to settle. A damage index was computed at each sensor for each tone burst frequency. The damage index is defined to be $(1 - \rho_{xy})$, where ρ_{xy} is the correlation coefficient between a segment of the reference and corresponding segment of the test signal. Segments are specified to include essentially the interval around the first arriving packet. Test signals are shown in Figure 23. The segment used for ρ_{xy} is approximately between 20 and 40 µsec.

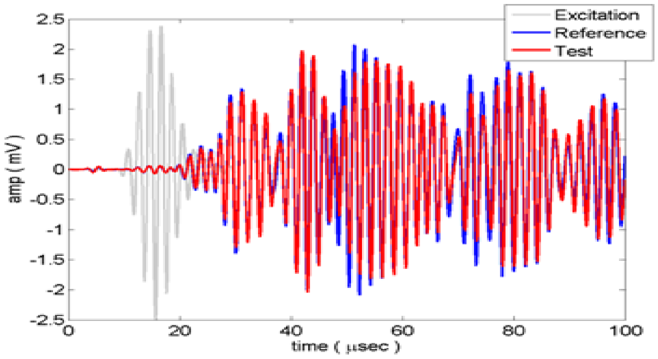


Figure 23. Excitation and Response Signals

Two methods were used to estimate the crack size. First, a linear regression model for mapping damage indices to visual crack size measurements was designed using data from the completed test. The data were randomly divided into training and test partitions across the experimental signal collections. For the six sensors and seven frequencies, 42 damage index values were computed at each CASE measurement (i.e., 42 damage index values per side). Feature selection was based on a stepwise regression procedure. The procedure involved iteratively fitting a series of multi-linear regression models to crack size measurements using different subsets of elements from the feature vector. The subset of features grows or shrinks based on the significance of a feature's contribution to the regression model. A feature is added to the subset only when its presence in the model improves the fit. Conversely, a feature is removed from the subset when its absence does not degrade the fit. The stepwise procedure terminates when the addition of any remaining feature does not improve the fit, and the removal of any previously selected feature degrades the fit.

Another estimation model was developed using an Artificial Neural Network (ANN). The ANN was trained using the damage index values and visual crack size measurements. As with the regression modeling, the data were randomly divided into training and test partitions across the experimental signal collections. Unlike the regression modeling, the ANN used all 42 damage index values. A diagram of the *state characterization* module is shown in Figure 24.

For this demonstration, two pairs of state estimation units were implemented (i.e., primary and secondary state estimations). The primary state estimation unit consists of regression model 1 and ANN 1 (see Figure 24). These estimates were used to provide the first state estimates (i.e., plausible states) to compete to enter working memory (i.e., conceptual system or global workspace). If both of these state estimates were unacceptable, then estimations from the secondary unit were used (i.e., regression model 2 and ANN 2) in the same manner. Note that these state estimation models are examples of the conscious framework principle *explicit representation* (see section 3.4.4).

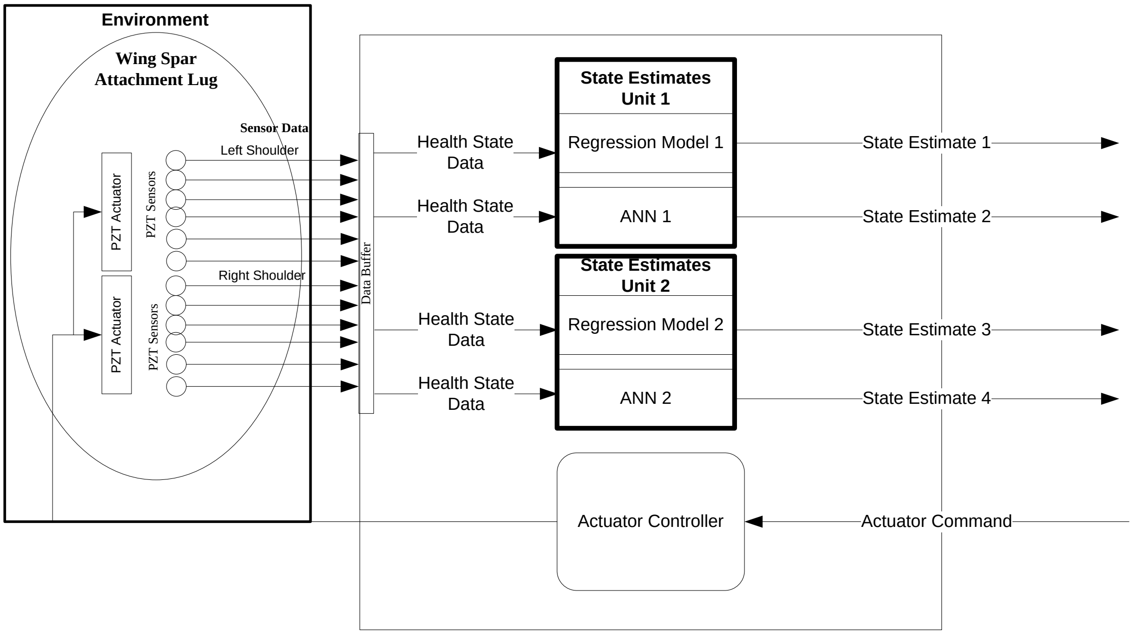


Figure 24. State Characterization Module of CASE as Demonstrated

4.5.2.3 State Selection

The *State Selection* module integrated information from *state characterization* and *environmental/operational processing* for selection of a current structural state (see Figure 25). The loads and cycle count information were used in an AFGROW model to predict the current crack size, while the regression and ANN estimation models provided estimates of the current crack size as described above. This is an example of the *evolving qualia* tenet of QUEST [28]. This process exploited a representation (i.e., model or simulation) to characterize the current state using past and predicted state estimates. The selected structural state was determined by a selection algorithm that used logic and agreement-based averaging.

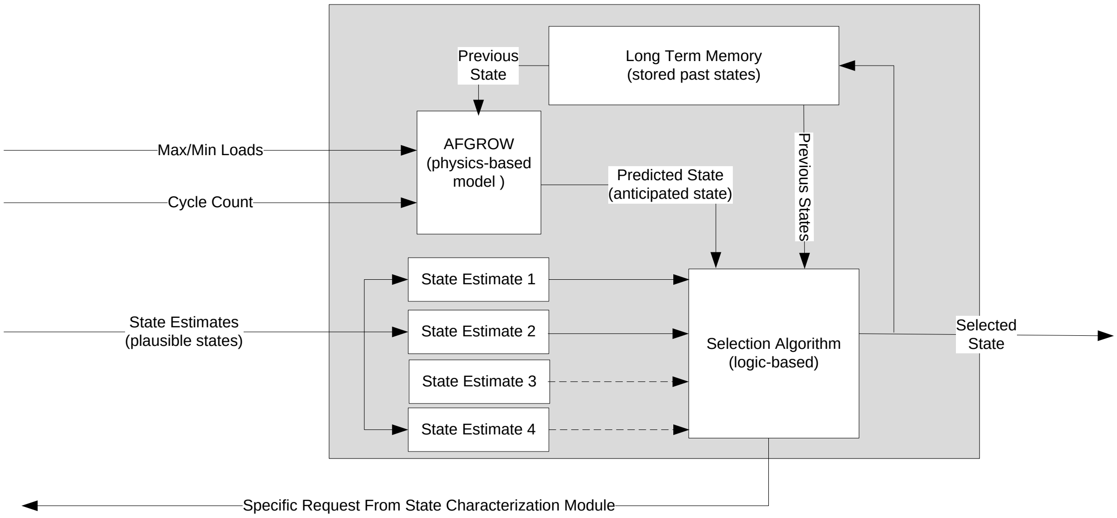

Previous State
Long Term Memory (stored past states)
Max/Min Loads
Cycle Count
AFGROW (physics-based model)
Predicted State (anticipated state)
Previous States
State Estimate 1
State Estimate 2
State Estimate 3
State Estimate 4
State Estimates (plausible states)
Selection Algorithm (logic-based)
Selected State
Specific Request From State Characterization Module

Figure 25. State Selection Module of CASE as Demonstrated

The logic incorporated into the selection algorithm was based on the fundamental premise that cracks do not get shorter. This logic ensures that current estimations must be equal to or greater than the previous selected state. If both the primary estimates violated this condition, they were rejected and removed from further consideration. However, if both the primary estimates were greater than the previous selected state and were within a certain percentage of each other, then their average was used as the current selected state. Averaging (i.e., combining) any of the state estimates illustrates the conscious framework attribute of *coalitions of neurons* [8]. If the estimates were not in agreement with each other, then each was checked individually for agreement with the AFGROW model predicted state. This logic can result in the selected state being an average of either one of the crack estimated states and the AFGROW predicted state, or an average of all three. If an agreement was not reached by this point, the *state selection* module would request two supplemental estimations, and the selection process would repeat. If an agreement was still not reached at the end of this phase, the selected state was defaulted to the AFGROW model predicted state. Selecting the model-based state estimate is an example of the *not measurable* QUEST tenet [28]. Figure 26 shows the state selection algorithm in detail.

<u>State Selection Algorithm:</u>

Four estimation states: $E_1(n)$, $E_2(n)$, $E_3(n)$, $E_4(n)$
Percentage value: ±%X Predicted state: $PS(n)$
Selected state: $SS(n)$ Previous state: $SS(n-1)$.

1.) If $E_1(n)$ & $E_2(n)$ are ≥ $SS(n-1)$) and $E_1(n)$ & $E_2(n)$ are within ±%X of each other {i.e. $E_1(n) - X * E_1(n) \leq E_2(n) \leq E_1(n) + X * E_1(n)$ and $E_2(n) - X * E_2(n) \leq E_1(n) \leq E_2(n) + X * E_2(n)$ } then $SS(n) = \frac{E_1(n)+E_2(n)}{2}$
2.) If $E_1(n)$ is within ±%X of $PS(n)$, then $SS(n) = \frac{E_1(n)+PS(n)}{2}$
3.) If $E_2(n)$ is within ±%X of $PS(n)$, then $SS(n) = \frac{E_2(n)+PS(n)}{2}$
4.) If $E_1(n)$ & $E_2(n)$ are within ±%X of $PS(n)$, then $SS(n) = \frac{E_1(n)+E_2(n)+PS(n)}{3}$
5.) If the first four steps do not yield the selected state, the architecture feedbacks to select two additional estimations or $E_3(n)= E_1(n)$, $E_4(n)= E_2(n)$ and repeat steps 1 to 4
6.) If the selected state is not determined by this point then $SS(n)$ is set equal to $PS(n)$

Figure 26. State Selection Algorithm

The demonstrated state selection algorithm was based on the assumption that the estimation techniques were able to detect a crack initiation of equal to or greater than 0.05”. If the estimations did not detect crack initiation, then the model prediction portion of the state selection was not activated, resulting in a continuous sequence of selected state crack lengths of 0”. Figure 27 plots the four estimated states, where primary estimations are *{REG(1),NN(1)}* and supplemental estimations are *{REG(2),NN(2)}*, the AFGROW predicted state (i.e., physics-based model estimation), and the visual crack measurements (i.e., truth data) from this experiment using laboratory data and a Simulink implementation of the state selection process. CASE detected crack initiation at 42,000 cycles. Prior to 42,000 cycles, no crack was detected; therefore, the predicted and selected states were 0”. At 42,000 cycles, the two primary estimates were neither in agreement with each other nor the predicted state. Hence, the algorithm requested the two supplemental estimations. These estimates were in agreement and determined crack initialization. The average of the supplemental estimates was used as the selected state, and is shown marked by X's in Table 3.

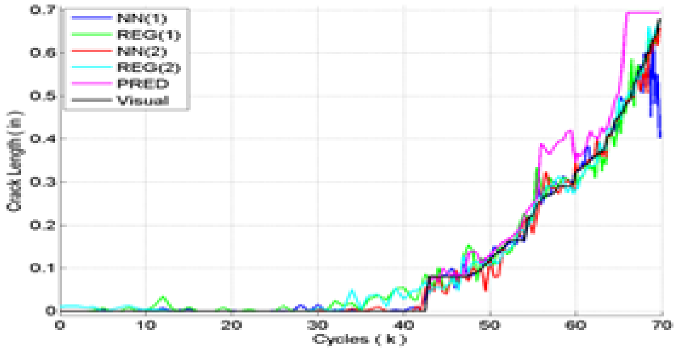


Figure 27. State Estimations and Visual Crack Size versus Cycles

Table 3 also shows that at the next state estimation cycle of 42,500 the two primary estimates were in agreement. Thus, no request for additional estimates was needed, and their average was used as the selected state. At cycle 43,000 the two primary estimates and the predicted state were in agreement, and their average was used as the selected state. Figure 28 shows the visual crack measurements and selected crack size resulting from applying the state selection algorithm for the lifetime of the lug.

Table 3. Estimates Selected for Averaging

Cycle (n)	*$NN_1(n)$*	*$REG_1(n)$*	*$NN_2(n)$*	*$REG_2(n)$*	*P(n)*
41,000					X
41,500					X
42,000			X	X	
42,500	X	X			
43,000	X	X			X

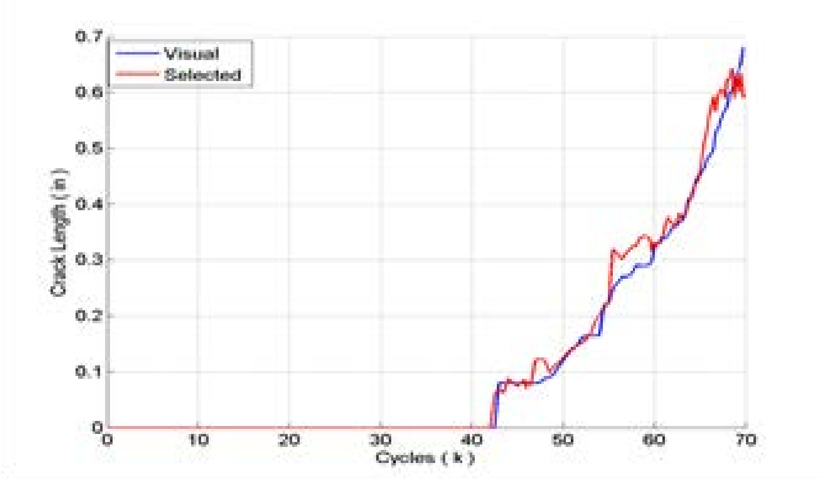


Figure 28. Selected State and Visual Crack Size versus Cycles

4.5.2.4 Action Selection

The Action Selection stage enhances the command's (i.e., user's) situational awareness by combining selected state information and mission goals into a common representation to enable effective and efficient operational decisions (see Figure 29).

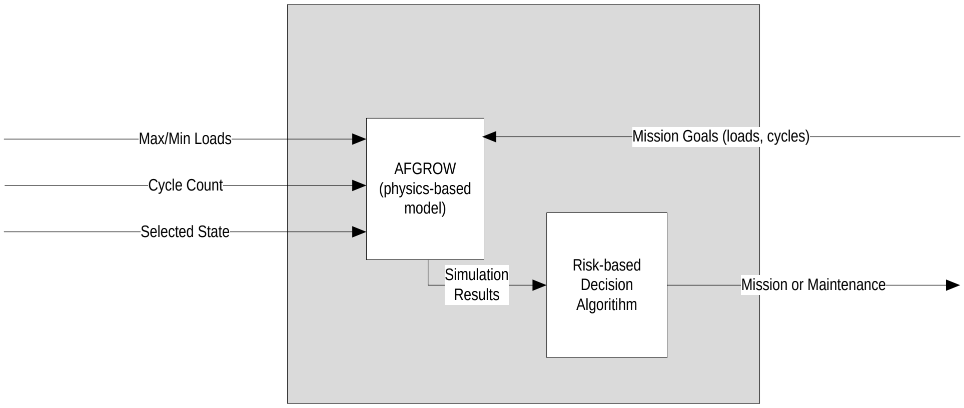

Max/Min Loads
Cycle Count
Selected State
AFGROW (physics-based model)
Mission Goals (loads, cycles)
Simulation Results
Risk-based Decision Algoritihm
Mission or Maintenance

Figure 29. Action Selection Module as Demonstrated

The time remaining before a crack reaches the predicted critical state can be estimated using AFGROW. Given the current crack size, along with material properties, part geometry, and loading information, AFGROW can estimate the number of cycles remaining before the crack reaches the critical crack state, resulting in failure. Figure 30 shows estimated crack sizes versus load cycles for selected peak loads. For example, consider entering the graph from the vertical axis at the crack length of 0.2” for the 1000 lbf peak load case. The AFGROW estimated cycle count corresponding to this crack length is approximately 11,800 cycles. Additionally, the number of cycles to failure has already been computed by AFGROW, and is 14,500 cycles the end point of the 1000 lbf curve. Therefore, the estimated remaining life of the component is found by subtraction to be 2,700 cycles (14,500–11,800).

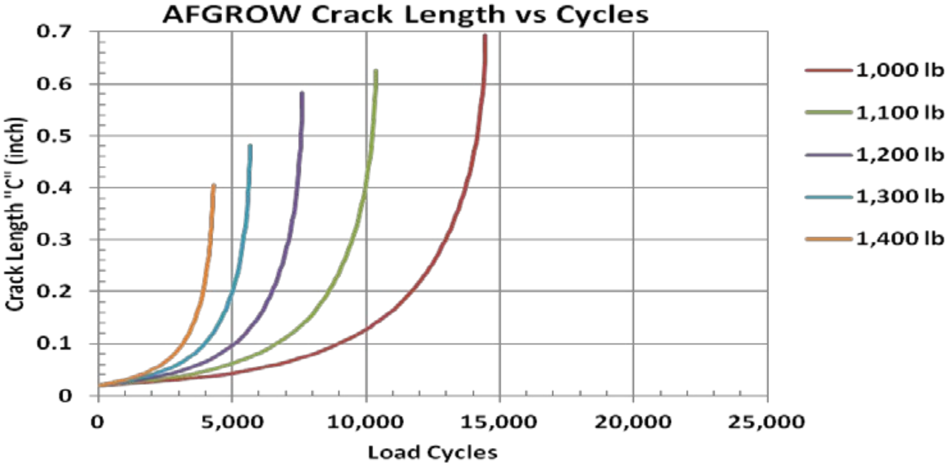


Figure 30. Model Predicted Crack Growth

For this demonstration, each mission was based on a cyclic load profile of 1,000 lbf with duration of 250 cycles or 50 flight hours (Cycles / 5 = flight hours). A typical risk chart was generated to enhance the command’s mission situational awareness. The risk chart was constructed with the vertical axis depicting the “Likelihood” and the horizontal axis representing the “Consequences.” The likelihood values were determined by calculating the difference between the estimated hours remaining before reaching a critical state given the current state and flight hours needed to complete the mission. Based on these calculated values, likelihood levels were assigned using the equations shown in Figure 31. Additionally, the consequences were determined by a random number generator assigning an integer value between one and four. The risk chart used during this demonstration is shown in Figure 32.

Level 1 = hours remaining – hours required > 1.90 * hours required.

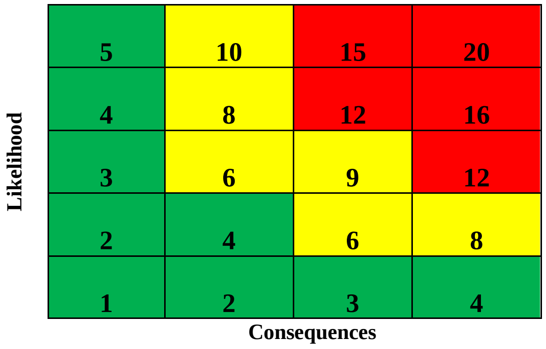
Level 2 = 1.90 * hours required > hours remaining – hours required > 1.70 * hours required.

Level 3 = 1.70 * hours required > hours remaining – hours required > 1.50 * hours required.

Level 4 = 1.50 * hours required > hours remaining – hours required > 1.30 * hours required.

Level 5 = 1.30 * hours required > hours remaining – hours required > 1.10 * hours required

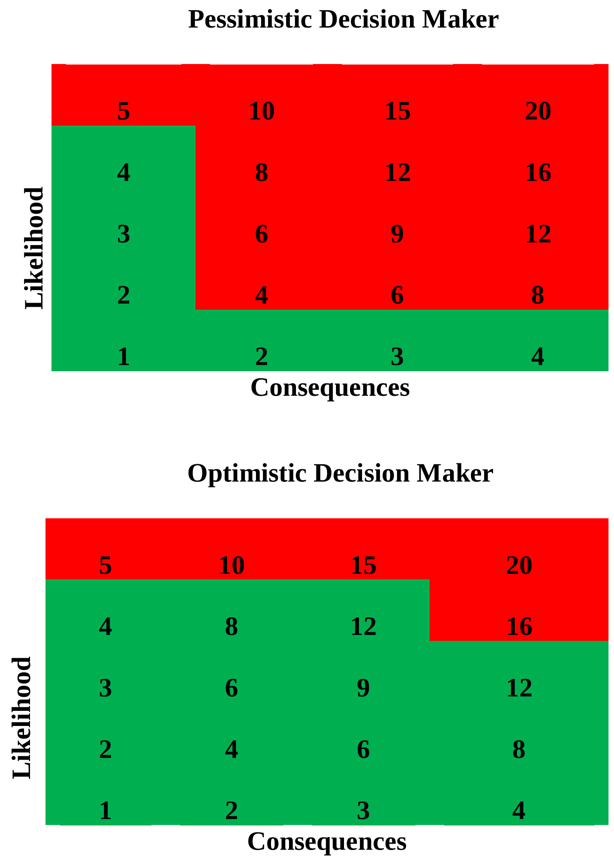
Figure 31. Risk Likelihood Equations


Likelihood
Consequences

5	10	15	20
4	8	12	16
3	6	9	12
2	4	6	8
1	2	3	4

Figure 32. Mission Risk Chart

Two categories of commanders were simulated during this demonstration to assess their impact on airframe operations. The first commander simulated was a pessimistic decision maker. This commander was risk averse and only decided to perform missions that have a high probability of success. Conversely, the second commander simulated was an optimistic decision maker and was willing to perform missions with a lower probability of success. The decision matrices for these commanders are shown in Figure 33.



Pessimistic Decision Maker

🟥 5	🟥 10	🟥 15	🟥 20
🟩 4	🟥 8	🟥 12	🟥 16
🟩 3	🟥 6	🟥 9	🟥 12
🟩 2	🟥 4	🟥 6	🟥 8
🟩 1	🟩 2	🟩 3	🟩 4

Likelihood
Consequences

Optimistic Decision Maker

🟥 5	🟥 10	🟥 15	🟥 20
🟩 4	🟩 8	🟩 12	🟥 16
🟩 3	🟩 6	🟩 9	🟩 12
🟩 2	🟩 4	🟩 6	🟩 8
🟩 1	🟩 2	🟩 3	🟩 4

Likelihood
Consequences

Figure 33. Decision Matrices

Data from the laboratory experiment was used in a Simulink simulation to evaluate the impact of three different operational philosophies. The experimental data was used to generate structural state information as described above. A total of 1,000 simulated lug life cycles were conducted comprising 280 missions per lifetime. Each mission within a given lifecycle produced a new consequence value via a random number generator. The likelihood values for each mission were calculated using the equations shown in Figure 30. Operational decisions were made using three different approaches. The first approach employed the current ASIP philosophy of repairing the airframe whenever a crack of any size was detected by the monitoring systems (i.e., reflexive).

This approach only required CASE to be used as a low-level SHM system. The next two methods were considered risk-based approaches. That is, weapon systems could continue to execute missions after a crack had been detected, depending upon the risk. To illustrate this approach, the pessimistic and optimistic decision makers were implemented during the simulation based on the decision matrix depicted in Figure 33. If the computed risk value corresponded to a green box, the decision was made to perform the next mission. Conversely, if the computed risk value corresponded to a red box, the decision was made to repair the airframe. The results of the simulation are shown in Figure 33 for each decision method.

The results indicate on average, the low-level SHM system will perform maintenance earlier than the risk-based approaches. On average, over the 1000 simulated runs, the low-level SHM system requested maintenance approximately 4,300 hours earlier than any of the risk-based decisions. However, the average differences between the risk-based decisions were much smaller. In fact, the results indicate the optimistic decision maker would recommend repair just 72 (12,888–12,816) hours beyond the pessimistic decision, as shown in Figure 34. This difference amounts to one additional mission since each mission has a 50 hour duration. A more quantitative comparison between ASIP and CASE is investigated in the next section.

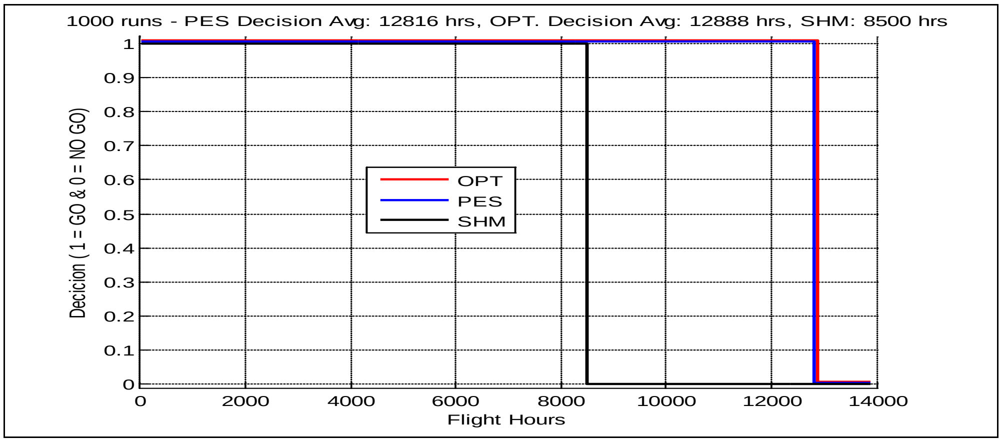


Figure 34. Simulation Results for 1,000 Lifetime Mission Decisions.

4.5.3 ASIP and CASE Comparison

A quantitative comparison of ASIP and CASE was performed using the experimental results and DoD's recommended metrics for assessing weapon systems operational effectiveness and efficiency [10]. The recommended metrics are as follows along with their corresponding definitions and formulas:

- *Materiel Reliability (M_R)* – a measure of the probability the system will perform without failure over the specific interval.

$$M_R = MTBM = \frac{Uptime}{\#\ Maintenance\ Actions}$$

Where *MTBM: Mean Time Between Maintenance*

- *Mean Down Time (MDT)* – the average total time required to restore an asset to its full operational capabilities.

$$MDT = \frac{Time\ per\ Maintenance * \#\ Maintenance\ Actions}{\#\ Maintenance\ Actions}$$

- *Materiel Availability (M_A)* – a measure of the percentage of time a system is operationally capable of performing an assigned mission at a given time, based on materiel condition.

$$M_A = \frac{MTBM}{MTBM+MDT}$$

- *Ownership Cost (OC)* – balances the sustainment solution by ensuring the O&S costs associated with materiel readiness are considered when making decisions.

$$OC = \#\, Maintenance\ Actions * Cost\ Per\ Maintenance\ Actions$$

For CASE/ASIP metric comparison, certain assumptions must be made regarding labor cost, maintenance down time, repair cost, etc. Table 4 shows the input assumptions for performing the metric calculations. These assumptions were selected from a recent cost benefit study performed by the Boeing Company for the Air Force Research Laboratory (AFRL) on a similar airframe component [23]. Also for CASE, only a single lug and the left side in the vertical direction over its operational lifetime was considered.

Using the assumptions and formulas, the efficiency metrics were calculated. First, the metrics using the ASIP process were calculated to serve as a baseline. Then, calculations were performed on a low-level SHM system and a high-level SHM system. The low-level SHM system requested a maintenance action whenever a crack of any size was detected. In contrast, the high-level SHM system called for maintenance based on risk using pessimistic and optimistic decision makers, as discussed in the previous section. Table 5 summarizes the calculated metrics for each monitoring approach investigated during this experiment.

Table 4. Simulation Input Assumptions and Parameters Description

Parameter	**Description**	**Wing Spar Attachment**
Number of locations	Quantity of locations or area per platform	2 corner cracks, left and right sides, in the vertical direction
Inspection Time	Time that covers accessing and inspecting all areas; assume both sides are inspected at the same time	200 labor hours
Inspection Interval	Time between inspections (ASIP)	860 flight hours
Normal Repair Time	Time to do repairs; assume these areas are already accessible due to inspection	200 labor hours
Normal Repair Additional Cost	Cost outside of repair labor, such as materials and support equipment; used for scheduled and unscheduled repairs.	> $100K (For a new clevis when a crack is found)
Extensive Repair Cost	Cost when crack requires extensive repair to bring the platform back to service	> $100K (For a new clevis when a crack causes fuel to leak)
Structure Replacement Time	Time to remove and replace a structure, when applicable	~200 labor hours
Labor Rate	Labor cost per hour to perform maintenance	$80.00/hour

Table 5. Summary of Metrics Calculations

	ASIP	Low level	High Level (Pessimistic)	High Level (Optimistic)	% Dif (Min)	% Dif (Max)
M_A	0.878	0.972	0.981	0.981	+10.7	+11.7
M_R (hrs)	860	8600	12816	12888	+900	+1399
OC ($)	192000	40000	40000	40000	-79	-79
MDT (hrs)	120	250	250	250	+108	+108

CASE produced improved values over the current ASIP process for three of the four evaluation metrics for all SHM systems investigated. That is, M_A and M_R increased by a minimum of 10.7 % and 900%, respectively. In addition, OC decreased by 79%. However, the *MDT* increased by 108%, which seems counterintuitive given the increase in M_R or *MTBM*. Details of these calculations are shown in Table 6.

After further examination, the *MDT* result is not so surprising. Since CASE only conducts maintenance when a repair is needed (not for inspections), and because the repair time is greater than the ASIP inspection time, the *MDT* increased. A large percentage of ASIP's down time is due to structural inspections. In fact, an actual structural repair would only be performed once during the ten scheduled maintenance intervals. The total down time for ASIP and CASE (low-level) for 8600 flight hours is shown in Figures 35 and 36, respectively. The graph indicates that ASIP total down time over this time interval is 1200 hours, and CASE total down time is only 250 hours (one maintenance request). Although the *MDT* for CASE is greater than ASIP's, its total down time is much less. In fact, CASE decreased the total down time by 79%, which explains the improvement in M_R or *MTBM*.

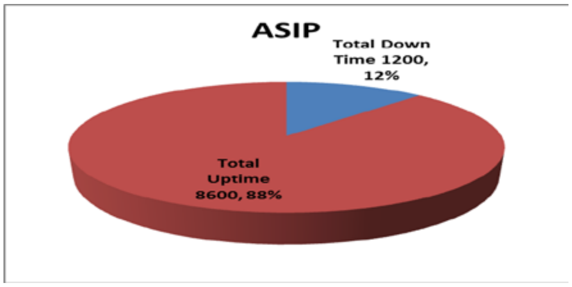


Figure 35. Total ASIP Down Time for 8600 Flight Hours

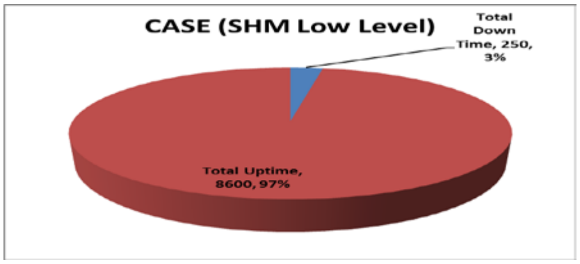


Figure 36. Total CASE Down Time for 8600 Flight Hours

Table 6. Metrics Calculation Results

ASIP Metrics Calculations			
Input Parameters		Calculations (ASIP)	
Maintenance Interval (hrs)	860	M_A	0.878
Time/inspection (hrs)	100	M_R (hrs)	860
Number of inspections	10	OC ($)	192000
Normal repair time (hrs)	200	MDT (hrs)	120
Number of repairs	1		
Man-hour labor	2		
Labor rate ($/hr)	80		
Operational time (hrs)	8600		
(CASE) Low-Level SHM Metrics Calculations			
Input Parameters		Calculations (SHM I)	
Maintenance Interval (hrs)	8600	M_A	0.972
Time/inspection (hrs)	100	M_R (hrs)	8600
Number of inspections	0	OC ($)	40000
Normal repair time (hrs)	250	MDT (hrs)	250
Number of repairs	1		
Man-hour labor	2		
Labor rate ($/hr)	80		
Operational time (hrs)	8600		
(CASE) High-Level SHM Metrics Calculations			
Input Parameters		Calculations (Pessimistic)	
Maintenance Interval (hrs)	12816	M_A	0.981
Time/inspection (hrs)	100	M_R (hrs)	12816
Number of inspections	0	OC ($)	40000
Normal repair time (hrs)	250	MDT (hrs)	250
Number of repairs	1		
Man-hour labor	2		
Labor rate ($/hr)	80		
Operational time (hrs)	12816		
(CASE) High-Level SHM Metrics Calculations			
Input Parameters		Calculations (Optimistic)	
Maintenance Interval (hrs)	12888	M_A	0.981
Time/inspection (hrs)	100	M_R (hrs)	12888
Number of inspections	0	OC ($)	40000
Normal repair time (hrs)	250	MDT (hrs)	250
Number of repairs	1		
Man-hour labor	2		
Labor rate ($/hr)	80		
Operational time (hrs)	12888		

Let's revisit the selected guiding principles discussed earlier to determine if they were used during the CASE demonstration. Table 3 above depicted a snapshot of the decisions made by the *state selection* module during the simulation. From this table, the guiding principles exercised can be determined via the operations of CASE. This process is illustrated next:

- ***Consciousness* –** At cycle 41,000 and 41,500, the predicted (i.e., model or simulation) estimates were selected. This condition only occurs if the primary and secondary estimates from the unconscious system (i.e., zombie system) are unacceptable. The model/simulation function is only utilized in the conscious mode.

- ***Explicit Representation* –** Cycle 42,500 is a good example of explicit representation. The output of each state estimation process (REG_1, NN_1, REG_2 and NN_2) is a crack length value, which makes them explicit representations. However, for this particular case (i.e., cycle 42,500), no further processing was required since the primary estimates were chosen (i.e., no request for addition data).

- ***Coalition of Neurons*** – Cycles 42,000 through 43,000 exhibit the coalition of neurons principle. Each estimate in the same row marked with an 'X' are averaged together to form a coalition.

- ***Unconscious*** – The unconscious mode is illustrated at cycle 42,500 since the primary estimates were chosen (REG_1 and NN_1) without interference from the conscious system.

- ***Attention/Binding*** – At cycle 42,000, CASE demonstrated the "top-down" attention mode because the secondary estimates were only active at the request of the conscious system. In this scenario, the secondary states were selected. Conversely, during cycle 42,500, the "bottom-up" attention mode was illustrated. This is evident because the primary estimates were selected without any directions from the conscious system. Binding occurred with all of the estimates since CASE uses a least two different processing methods (e.g., REG_1 and NN_1) before deciding on a state. The REG and NN models could be regarded as observing the same object from different aspects. The REG model is linear method, while the NN is a non-linear process. In addition, the REG and NN models do not use the same features as described in section 4.5.2.2.

Table 1 above depicted the correlation between the conscious framework attributes and the SIT/Quest tenets. From this relationship, it is safe to infer that all the SIT/Quest tenets in Table 1 were utilized since their corresponding attributes were exhibited.

4.5.4 Summary

CASE is demonstrated via an SHM application. A representative airframe component is used in a simulated fatigue experiment to compare CASE with the current ASIP process. CASE and ASIP processes are described in detail. The DOD metrics for maintaining aircraft are used for comparing CASE with ASIP.

V. Conclusion

5.1 Overview

This chapter discusses the contributions of this research and provides recommendations for future work. Furthermore, it describes how the research objectives were achieved and concludes with a summary of the research.

5.2 Contributions

The current research makes contributions in the area of machine conscious architectures. It culminated in a design, development and demonstration of a general purpose architecture for information processing and decision making (i.e., CASE). The motivation behind CASE is to engineer a solution for applications that have not been successfully addressed through traditional computational intelligence techniques. This novel architecture was designed to mimic key characteristics of human cognition. Specifically, it incorporates particular features of the unconscious and conscious processes of human cognition. Although several researchers have developed architectures to mimic specific functionalities [2] [4] [25] [30] of consciousness, the "hard problem" of creating artificial phenomenal states has not been adequately addressed during these investigations.

The uniqueness in this research is that CASE incorporates guiding principles for consciousness recommended by [8] [22] [28] to include key characteristics of both functional and phenomenal behavior. The current research demonstrated each of the selected key characteristics of consciousness via CASE using software simulation. These guiding principles do not ensure the generation of phenomenal states; however, by using the recommended guidance, a framework emerges to further the investigation of MC and phenomenal states.

CASE was demonstrated in a selected application to determine its engineering advantages. It was used in an SHM application to autonomously monitor the integrity of a flight critical airframe component, and it automatically generated a timely maintenance recommendation when unacceptable cracking was detected. Metrics computed from experimental results demonstrated that using CASE is more effective and efficient than the currently employed maintenance approach (i.e., ASIP). Over the lifetime of the investigated component, operational availability increased by a minimum of 10.7%, operational cost decreased by 79%, and maintenance intervals (i.e., *MTBM*) increased by a minimum of 900%.

5.3 Future Work

The research field of MC is in its infancy, and therefore, the opportunity for future research is bountiful. However, developing a standard framework (i.e., architecture) to investigate the "hard problem" of phenomenal states (i.e., qualia) is paramount. Recommendations of future work to address this critical problem are given below using the current research as the point of departure.

5.3.1 Tenets/Attributes of Consciousness

The current research used five of the recommended conscious framework attributes and their associated functional and behavioral tenets to illustrate in CASE. Future research needs to incorporate more of these recommended tenets into a single architecture for continued advancement in creating a general purpose MC framework.

This approach will provide structure for researchers to further the investigation of MC, specifically phenomenal states. A complete list of the QUEST tenets is provided at [26]. It is recommend that the next tenets integrated into CASE be chosen from the QUEST tenets of qualia since they attempt to capture the characteristics of phenomenal states (i.e., qualia).

5.3.2 Engineering Applications

Additional demonstrations are required to further mature the MC framework and investigate the engineering advantages enabled by these systems. Rogers et al. provided a list of diverse problems applicable to MC solutions. Examining the results from these experiments will allow for the adding and/or removing of tenets as needed, enabling the establishment of design criteria for developing MC systems since one does not currently exist.

5.3.3 Evaluation Criteria

Lastly, MC research needs to focus on creating evaluation criteria for architectures/frameworks. Currently, there is no way to determine whether or not an MC system has been successfully developed [21] [25]. Once a design standard has been established, the accompanying assessment process should ensue to determine whether or not the MC framework was successfully designed. Again, this process will create a structure for MC researchers to follow.

5.4 Summary

This research designed, developed and demonstrated an MC architecture, CASE, conforming to key guiding principles of consciousness. A detailed design of CASE was described, along with illustrations of the selected key features of consciousness. CASE was then demonstrated in an SHM application to determine if it provided an engineering advantage compared to the current technique.

Bibliography

[1] Aleksander, I. (2009). Designing Conscious Systems. *Cognitive Computing, 1*(1), 22-28.

[2] Arrabales, R., Ledezma, A., & Sanchis, A. (2010). Toward the Generation of Visual Qualia in Artificial Cognitive Architectures. *BICS 2010 Conference on Brain-Inspired Cognitive Systems* (pp. 1-14). Brain Inspired Cognitive Systems (BICS).

[3] Baars, B. J. (1997). In the Theater of Consciousness: Global Workspace Theory, A Rigorous Theory of Consciousness. *Journal of Consciousness Studies 4*, 292-309.

[4] Baars, B. J., & Franklin, S. (2007). An Architectural Model of Conscious and Unconscious Brain Functions: Glodal Workspace Theory and IDA. *Neural Networks*, 955-961.

[5] Balageas, D. (2006). Introduction to Structural Health Monitoring. In D. Balageas, C.-P. Fritzen, & A. Guemes, *Structural Health Monitoring* (pp. 13 - 44). Wiley.

[6] Birrer, B. D. (2010). Developing a Qualia-Based Multi-Agent Architecture for Used in Malware Detection. *Ph.D. Dissertation*, AFIT-DCS-ENG-10-01.

[7] Clark, A. (2002). Is Seeing All it Seems? Action, reason and the grand illusion. *Journal of Consciousness Studies, 9*, 181-202.

[8] Crick, F., & Koch, C. (2003). A Framework for Consciousness. *Nature Neuroscience, 6*, 119-126.

[9] Department of Defense. (2005). *Aircraft Structural Integrity Program General Guidelines.*

[10] Department of Defense. (2008). *Condition-Based Maintenance Plus DoD Guidebook.*

[11] Derriso, M. M., DeSimio, M. P., & McCurry, C. D. (2012). Global Workspace Theory Inspired Architecture for Autonomous Structural Health Monitoring. *NEACON.* Dayton: IEEE.

[12] Derriso, M. M., Olson, S. E., & DeSimio, M. P. (2009). Military Aircraft. In C. Boller, F.-K. Chang, & Y. Fujino (Eds.), *Encyclopedia of Structural Health Monitoring* (Vol. 1). Wiley Blackwell.

[13] Derriso, M. M., Olson, S. M., & Pratt, D. M. (2007). Why Are There Few Fielded SHM Systems for Aerospace Structures. *International Workshop on Structural Health Monitoring*, (pp. 44 - 55). Stanford.

[14] Doebling, S. F., Farrar, C. R., Prime, M. B., & Shevitz, D. W. (May 1996). *Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in Their Vibration Characteristics: A Literature Review*. Los Alamos, NM: Los Alamos National Laboratory (LA-13070-MS).

[15] DOT/FAA/AR-MMPDS-01. (Jan 2003). *Metallic Materials Properties Development and Standardization (MMPDS).* MMPDS.

[16] Dube, T. E. (2011). A Novel Malware Target Recogniton Architecture for Enhanced Cyberspace Situation Awareness. *Ph.D. Dissertation*, AFIT/DCE/ENG/11-07.

[17] Eckbreth, A. C., & Saff, C. R. (2011). *Sustaining Air Force Aging Aircraft into the 21st Century.* United States Air Force, Scientific Advisory Board (SAB). SAB-TR-11-01.

[18] Eletter, S. F., & Yaseen, S. G. (2010). Applying Neural Networks for Loan Decisions in the Jordanian Commercial Banking System. *International Journal of Computer Science and Network Security, 10*(1), 209-214.

[19] Farrar, C. R., & Worden, K. (2007). An Introduction to Structural Health Monitoring. *Philosophical Transactions of the Royal Society, A*(365), 303-315.

[20] Ihn, J.-B., & Chang, F. k. (2008). Pitch-Catch Active Sesning Methods in Structural Health Monitoring for Aircraft Structures. *Structural Health Monitoring, 7*, 5-19.

[21] Long, L. N., & Kelley, T. D. (2009). The Requirements and Possibilities of Creating Conscious Systems. *AIAA.* Seattle: AIAA.

[22] Morsella, E. (2005). The Function of Phenomenal States: Supramodular Interaction Theory. *Psychological Review, 112*(4), 1000-1021.

[23] Poblete, J., Torng, T., Dunne, J., Ihn, J., McFarland, J., & Reid, P. (2010 Limited Distribution). *Cost Benefit Analysis of SHM for F-15 Hot Spots.* The Boeing Company.

[24] Raghavan, A., & Cesnik, C. (2007). Review of Guided Wave Structural Health Monitoring. *The Shock and Virbration Digest, 39*(2), 91-114.

[25] Ramamurthy, U., & Franklin, S. (2009). Resilient Architectures to Facilitate Both Functional Consciousness and Phenomenal Consciousness in Machines. *International Journal of Machine Consciousness, 1*(2), 243-253.

[26] Rogers, S. K., & Adam, R. (2009, August). *Quale, LLC*. Retrieved February 7, 2013, from Quest Group's Blog: http://qualellc.wordpress.com/

[27] Rogers, S. K., Kabrisky, M., Bauer, K., & Oxley, M. E. (2003). Computing Machinery and Intelligence Amplification. In D. B. Fogel, & C. J. Robinson (Eds.), *Computational Intelligence: The Experts Speak* (pp. 25-44). IEEE.

[28] Rogers, S. K., Sadowski, C., Bauer, K. W., Qxley, M. E., Kabrisky, M., Rogers, A., et al. (2008). The Life and Death of ATR/Sensor Fusion and Hope for Resurrection. In F. A. Sadjadi, & A. Mahalanobis (Ed.), *SPIE, Automatic Target Recognition XVIII*, *6967.* Orlando.

[29] Shanahan, M. (2006, June). A Cognitive Architecture that Combines Internal Simulation with a Global Workspace. *Consciousness and Cognition, 15*(2), 433-449.

[30] Shanahan, M., & Baars, B. (2005, December). Applying the Global Workspace Theory to the Frame Problem. *International Journal of Cognition Science, 98*(2), 157-176.

[31] Sloman, A., & Chrisley, R. (2003). Virtural Machines and Consciousness. (O. Holland, Ed.) *Journal of Consciousness Studies, 10*(4-5), 133-172.

[32] Sohn, H., Farrar, C. R., Hemez, F. M., Shunk, D. D., Stinemates, D. W., Nadler, B. R., et al. (2004). *A Review of Structural Health Monitoring Literature: 1996-2001.* Los Alamos.

[33] Tong, Y. C. (2001). *Literature Review on Aircraft Structural Risk and Reliability Analysis.* Airframes and Engines Division. Defence Science and Technology Organisation (DSTO).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*
25 February 2013

2. REPORT TYPE
Doctoral Dissertation

3. DATES COVERED *(From – To)*
28 September 2009 – 25 February 2013

4. TITLE AND SUBTITLE
Machine Conscious Architecture For State Exploitation and Decision Making

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Mark M. Derriso, DR-03, USAF

5d. PROJECT NUMBER
N/A

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/ENG)
2950 Hobson Way, Building 640
WPAFB OH 45433-8865

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT-ENG-DS-13-M-01

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Intentionally Left Blank

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.

13. SUPPLEMENTARY NOTES
This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT

This research addressed a critical limitation in the area of computational intelligence by developing a general purpose architecture for information processing and decision making. Traditional computational intelligence methods are best suited for well-defined problems with extensive, long-term knowledge of the environmental and operational conditions the system will encounter during operation. These traditional approaches typically generate quick answers (i.e., reflexive responses) using pattern recognition methods. Most pattern recognition techniques are static processes which consist of a predefined series of computations. For these pattern recognition approaches to be effective, training data is required from all anticipated environments and operating conditions. The proposed framework, Conscious Architecture for State Exploitation (CASE), is a general purpose architecture designed to mimic key characteristics of human information processing. CASE combines low- and high-level cognitive processes into a common framework to enable goal-based decision making. The CASE approach is to generate artificial phenomenal states (i.e., generate qualia = consciousness) into a shared computational process to enhance goal-based decision making and adaptation. That is, this approach allows for the appropriate decision and corresponding adaptive behavior as the goals and environmental factors change.

15. SUBJECT TERMS
Machine Consciousness, Computational Intelligence, Architectures

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Richard A. Raines, Ph.D
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	84	(937) 255-6565, x 4278